

Grado Ingeniería Informática

2016-2017

Trabajo Fin de Grado

“USO DE TELÉFONOS IOS PARA CONTESTAR PREGUNTAS EN CLASES PRESENCIALES”

César Irnán Sillero

Tutor/es

Ángel García Olaya

Leganés, octubre de 2017



[Incluir en el caso del interés de su publicación en el archivo abierto]

Esta obra se encuentra sujeta a la licencia Creative Commons
Reconocimiento – No Comercial – Sin Obra Derivada

Agradecimientos

Llegados al final de la carrera me gustaría tener unas palabras de agradecimiento para con las personas que han sido un pilar fundamental a lo largo de mi etapa universitaria.

Para empezar, a mis padres, sin su apoyo y su constancia sobre mí, no habría sido capaz de ser nada de lo que soy hoy en día.

A mi novia, Bea, motivo por el cual lucho cada día para que tengamos una vida juntos y qué mejor que tener un novio con título.

A Pablo, compañero, amigo y hermano, sin ti esas tardes de zulo habrían sido catastróficas.

A todos ellos, muchísimas gracias por pertenecer a mi vida.

Índice

AGRADECIMIENTOS.....	1
ÍNDICE DE ILUSTRACIONES	4
ÍNDICE DE TABLAS	5
INTRODUCCIÓN.....	8
DESCRIPCIÓN DEL PROBLEMA	8
MOTIVACIÓN	11
OBJETIVOS.....	12
PLANIFICACIÓN.....	13
ESTRUCTURA DEL DOCUMENTO	14
ESTADO DEL ARTE	15
SISTEMA OPERATIVO	15
<i>iOS.....</i>	<i>18</i>
<i>Xcode</i>	<i>20</i>
SISTEMAS SIMILARES	20
<i>Socrative</i>	<i>20</i>
<i>BlackBoard Learn</i>	<i>20</i>
<i>EduClick.....</i>	<i>21</i>
TABLA COMPARATIVA	21
ANÁLISIS, DISEÑO, IMPLEMENTACIÓN E IMPLANTACIÓN	23
INTRODUCCIÓN	23
ANÁLISIS	24
<i>Casos de uso.....</i>	<i>24</i>
<i>Especificación de los requisitos</i>	<i>30</i>
<i>Matriz de trazabilidad.....</i>	<i>46</i>
DISEÑO	47
<i>Arquitectura del sistema.....</i>	<i>47</i>
<i>Normas de diseño y construcción y estándares</i>	<i>48</i>
<i>Entorno tecnológico</i>	<i>48</i>
<i>Conectividad</i>	<i>49</i>
<i>Diseño de clases.....</i>	<i>49</i>
<i>Protocolo de comunicaciones</i>	<i>56</i>

<i>Funcionamiento del sistema</i>	62
<i>Interfaces</i>	64
<i>Diagrama de navegabilidad</i>	67
IMPLEMENTACIÓN	68
<i>Implementación del sistema</i>	68
<i>Problemas encontrados</i>	68
IMPLANTACIÓN	69
EVALUACIÓN	70
DESCRIPCIÓN DE LAS PRUEBAS	71
PRUEBAS DE VALIDACIÓN	72
PRUEBAS DE SISTEMA	73
PRUEBAS DE RENDIMIENTO.....	74
TRAZABILIDAD ENTRE PRUEBAS Y REQUISITOS.....	75
PLANIFICACIÓN Y PRESUPUESTO	76
PLANIFICACIÓN.....	76
PRESUPUESTO	76
<i>Costes de recursos humanos</i>	76
<i>Coste de licencia</i>	77
<i>Costes de hardware</i>	77
<i>Costes indirectos</i>	78
<i>Costes totales</i>	79
CONCLUSIONES Y TRABAJOS FUTUROS.....	80
CONCLUSIONES Y VALORACIÓN PERSONAL	80
TRABAJOS FUTUROS	82
ANEXO I	83
MANUAL USUARIO APLICACIÓN	83
<i>Visualizar histórico</i>	83
<i>Realizar test</i>	84
REFERENCIAS.....	87
BIBLIOGRAFÍA	87
GLOSARIO DE TÉRMINOS	88

Índice de ilustraciones

ILUSTRACIÓN 1 - PLANIFICACIÓN DEL PROYECTO (GANTT).....	13
ILUSTRACIÓN 2 - CUOTA MERCADO MÓVIL 2Q17	17
ILUSTRACIÓN 3 - ADOPCIÓN IOS POR VERSIÓN.....	18
ILUSTRACIÓN 4 - DIAGRAMA CASOS DE USO ALUMNO	24
ILUSTRACIÓN 5 - DIAGRAMA CASOS DE USO PROFESOR.....	25
ILUSTRACIÓN 6 - ARQUITECTURA DEL SISTEMA	47
ILUSTRACIÓN 7 - DIAGRAMA DE CLASES	50
ILUSTRACIÓN 8 - CLASE AppDELEGATE	51
ILUSTRACIÓN 9 - CLASE CLIENT	51
ILUSTRACIÓN 10 - CLASE MARK	52
ILUSTRACIÓN 11 - CLASE OPTION	52
ILUSTRACIÓN 12 - CLASE QUESTION	53
ILUSTRACIÓN 13 - CLASE PROTOC	54
ILUSTRACIÓN 14 - CLASE QUIZ.....	55
ILUSTRACIÓN 15 - CLASE HELPER	56
ILUSTRACIÓN 16 - SUB-PROTOCOLO ENVÍO CREDENCIALES.....	57
ILUSTRACIÓN 17 - SUB-PROTOCOLO ENVÍO DE OPCIONES	58
ILUSTRACIÓN 18 - SUB-PROTOCOLO ENVÍO DE PREGUNTAS.....	59
ILUSTRACIÓN 19 - DIAGRAMA DE SECUENCIA - EJECUTAR TEST.....	62
ILUSTRACIÓN 20 - DIAGRAMA DE SECUENCIA - HISTÓRICO DE CALIFICACIONES.....	63
ILUSTRACIÓN 21 - INTERFAZ – LAUNCHSCREEN	64
ILUSTRACIÓN 22 - INTERFAZ – HOME.....	65
ILUSTRACIÓN 23 - INTERFAZ – CONEXIÓN	65
ILUSTRACIÓN 24 - INTERFAZ – TEST	66
ILUSTRACIÓN 25 - INTERFAZ - HISTÓRICO CALIFICACIONES.....	66
ILUSTRACIÓN 26 - DIAGRAMA DE NAVEGABILIDAD	67
ILUSTRACIÓN 27 - MANUAL HISTÓRICO RESULTADOS	83
ILUSTRACIÓN 28 - MANUAL CONEXIÓN	84
ILUSTRACIÓN 29 - MANUAL CONTRASEÑA	85
ILUSTRACIÓN 30 - MANUAL TEST.....	85
ILUSTRACIÓN 31 - MANUAL FINALIZACIÓN	86

Índice de tablas

TABLA 1 - COMPARATIVA SISTEMAS OPERATIVOS MÓVILES	17
TABLA 2 - TABLA COMPARATIVA SOLUCIONES SIMILARES	22
TABLA 3 - PLANTILLA DE CASOS DE USO	26
TABLA 4 - CU01 – IMPORTAR TEST	27
TABLA 5 - CU02 – IMPORTAR ALUMNO	27
TABLA 6 - CU03 – CREAR TEST	27
TABLA 7 - CU04 – MODIFICAR TEST	28
TABLA 8 - CU05 – REALIZAR TEST	28
TABLA 9 - CU06 – RECIBIR&EXPORTAR NOTAS	28
TABLA 10 - CU07 – VISUALIZAR FEEDBACK.....	29
TABLA 11 - CU08 – CONEXIÓN E IDENTIFICACIÓN	29
TABLA 12 - CU09 – REALIZAR TEST	29
TABLA 13 - CU10 – ENVIAR EVALUACIÓN.....	30
TABLA 14 - CU11 – VISUALIZAR FEEDBACK.....	30
TABLA 15 - CU12 – HISTÓRICO RESULTADOS	30
TABLA 16 - PLANTILLA DE REQUISITOS	31
TABLA 17 - RSFNC-001.....	32
TABLA 18 - RSFNC-002.....	33
TABLA 19 - RSFNC-003.....	33
TABLA 20 - RSFNC-004	33
TABLA 21 - RSFNC-005.....	33
TABLA 22 - RSFNC-006.....	34
TABLA 23 - RSFNC-007.....	34
TABLA 24 - RSFNC-008.....	34
TABLA 25 - RSFNC-009.....	34
TABLA 26 - RSFNC-010.....	35
TABLA 27 - RSFNC-011.....	35
TABLA 28 - RSFNC-012.....	35
TABLA 29 - RSFNC-013.....	35
TABLA 30 - RSFNC-014.....	36
TABLA 31 - RSFNC-015.....	36
TABLA 32 - RSNFN-001	36
TABLA 33 - RSNFN-002	36
TABLA 34 - RSNFN-003	37
TABLA 35 - RSNFN-004	37
TABLA 36 - RSRND-001	37

USO DE TELÉFONOS IOS PARA CONTESTAR PREGUNTAS EN CLASES PRESENCIALES

TABLA 37 - RSRND-002	37
TABLA 38 - RSRND-003	38
TABLA 39 - RSITF-001.....	38
TABLA 40 - RSITF-002.....	38
TABLA 41 - RSITF-003.....	38
TABLA 42 - RSITF-004.....	39
TABLA 43 - RSITF-005.....	39
TABLA 44 - RSITF-006.....	39
TABLA 45 - RSRSC-001	39
TABLA 46 - RSRSC-002	40
TABLA 47 - RSRSC-003	40
TABLA 48 - RSRSC-004	40
TABLA 49 - RSRSC-005	40
TABLA 50 - RSCMP-001.....	41
TABLA 51 - RSCMP-002.....	41
TABLA 52 - RSCMP-003.....	41
TABLA 53 - RSCMP-004.....	41
TABLA 54 - RSCMP-005.....	42
TABLA 55 - RSACP-001.....	42
TABLA 56 - RSACP-002.....	42
TABLA 57 - RSACP-003.....	42
TABLA 58 - RSACP-004.....	43
TABLA 59 - RSACP-005.....	43
TABLA 60 - RSACP-006.....	43
TABLA 61 - RSSGD-001.....	43
TABLA 62 - RSSGD-002.....	44
TABLA 63 - RSSGD-003.....	44
TABLA 64 - RSSGD-004.....	44
TABLA 65 - RSSGD-005.....	44
TABLA 66 - RSQTY-001.....	45
TABLA 67 - RSQTY-002.....	45
TABLA 68 - RSQTY-003.....	45
TABLA 69 - RSQTY-004.....	45
TABLA 70 - TRAZABILIDAD ENTRE REQUISITOS FUNCIONALES Y CASOS DE USO	46
TABLA 71 - ESPECIFICACIÓN HASH DE CONFIGURACIÓN	60
TABLA 72 - SUB-PROTOCOLO ENVÍO TEST COMPLETO	61
TABLA 73 – PLANTILLA PRUEBAS	71

USO DE TELÉFONOS IOS PARA CONTESTAR PREGUNTAS EN CLASES PRESENCIALES

TABLA 74 - PRVAL-01	72
TABLA 75 - PRVAL-02	72
TABLA 76 - PRSYS-01	73
TABLA 77 - PRSYS-02	73
TABLA 78 - PRRND-01	74
TABLA 79 - TRAZABILIDAD PRUEBAS Y REQUISITOS	75
TABLA 80 - COSTES RECURSOS HUMANOS	77
TABLA 81 - COSTES DE SOFTWARE	77
TABLA 82 - COSTES HARDWARE	78
TABLA 83 - COSTES INDIRECTOS	78
TABLA 84 - COSTES TOTALES	79

Introducción

En este apartado del documento se detalla el proceso de análisis, diseño e implementación, así como la gestión y costes del proyecto para implementar una solución Mobile, haciendo uso de los teléfonos inteligentes o Smartphones, para contestar preguntas en clases presenciales.

Como anexo al documento se incluirá un manual de usuario de la aplicación, con una explicación detallada del uso de la misma.

Descripción del problema

Las nuevas tecnologías y su avance están marcando una época de transición, en la que cada vez se utilizan menos los medios convencionales para adoptar nuevas formas de comunicación, entretenimiento y aprendizaje.

En el ámbito educativo, este proceso de renovación es, asimismo, palpable. En el sistema educativo actual, basado en calificaciones obtenidas a través de pruebas de conocimientos periódicas, para así mantener al alumno obligado a trabajar la asignatura de manera continuada diaria, semanal o mensualmente, no certifica que un alumno haya comprendido un tema o un concepto, si no que únicamente evalúa un conocimiento en un momento determinado, por lo tanto, se hace necesario un cambio de paradigma educacional, en el que nos encontramos en proces.

Estas pruebas de concepto tienen diferentes tipologías: exámenes parciales, preguntas teóricas, pruebas conceptuales, exámenes tipo test, o cualquier tipo de prueba que el profesor considere necesaria para validar los conocimientos adquiridos sobre la materia.

Este proyecto, se centrará en los exámenes tipo test, normalmente realizados en formato físico (utilizando bolígrafo y papel) lo que hace necesaria una corrección manual o asistida (utilizando algún software de reconocimiento óptico de caracteres OCR). De esta complicación surge la oportunidad de crear una herramienta capaz de corregir los test de manera inmediata.

En la Universidad Carlos III de Madrid se implantó hace algunos años una plataforma de apoyo virtual a la enseñanza denominada “Aula Global” basado en el popular LMS (sistema de gestión del aprendizaje) Moodle. En Aula Global se ofrece la posibilidad de realizar formularios y cuestionarios, sin embargo, no satisface la mayor restricción planteada por un profesor en un curso presencial, la asistencia. Puesto que Aula Global ofrece la posibilidad de conectarte desde cualquier dispositivo que tenga conexión a internet, lo cual no implica la asistencia física del alumno en el aula. Esta salvedad puede ser mitigada realizando los exámenes físicamente en un aula de informática, pero esto impone una nueva restricción, no se pueden realizar test en las aulas donde se imparten las clases magistrales.

Debido a esta restricción, hace tiempo surgió EduClick, un sistema de mandos de respuesta interactiva aplicado a la docencia. Esta solución utiliza tanto software como hardware, el cual, usando un pequeño mando a distancia, vinculado a un dispositivo receptor, permite responder de manera individual a cada una de las preguntas de un examen tipo test. A simple vista parece un sistema eficaz, pero pronto, algunos profesores se mostraron reacios al uso de este método, puesto que, o bien desconocían el sistema, o no tenían acceso, puesto que existen unidades limitadas, o no había suficientes dispositivos para todos los alumnos, o no tenían el conocimiento técnico suficiente para realizar dicha prueba. Si a estos desconocimientos y restricciones, se le añade la sencillez de realizar un formulario test en un editor de texto y realizar fotocopias para repartir a todos los alumnos, la utilización de EduClick fue disminuyendo hasta casi el desconocimiento, lo que supone un gasto en licencias y material para caer al olvido. Si a parte de todo esto, la mayor facilidad que ofrece EduClick, que es la integración con sistemas terceros como Moodle, depende de actualizaciones, al final se observa que es un sistema obsoleto y caro de mantener.

Por lo tanto, una vez analizada la situación actual, en 2015 se planteó la posibilidad de que la Universidad Carlos III de Madrid crease su propia aplicación para responder a preguntas en clase usando un dispositivo móvil. Puesto que hoy en día la gran mayoría de alumnos dispone de Smartphone se llegó a la conclusión de que la mejor opción sería crear una arquitectura cliente servidor en la que el profesor pudiese crear pruebas teóricas tipo test y que cada alumno, independientemente del terminal que poseyera, respondiese a cada pregunta de manera ágil y cómoda. En sendos Trabajos Fin

USO DE TELÉFONOS IOS PARA CONTESTAR PREGUNTAS EN CLASES PRESENCIALES

de Grado se crearon una aplicación de escritorio para el profesor y aplicaciones móviles para teléfonos Android y Windows.

Motivación

Tras terminar el periodo académico y con el gran avance tecnológico que suponen los dispositivos móviles, este proyecto ofrece la posibilidad de mejorar el sistema de evaluación tanto a alumnos como a profesores, mejorando y poniéndolo al servicio de estos últimos. El objetivo es acercar esta tecnología a todas las áreas docentes, ya que no harán falta conocimientos técnicos profundos para realizarlos. Por tanto, la finalidad de este método, es simplificar el proceso de evaluación y reducir los tiempos de corrección, liberando al profesorado de una carga de trabajo que se puede realizar de manera automática.

Otro punto de vista es el del avance tecnológico, el cómo desde una idea inicial, los exámenes tipo test en formato físico, se puede, mediante un estudio de necesidades y una revisión tecnológica del proceso, llegar a crear, evolucionar y mejorar un método ya existente.

Al ser una aplicación móvil, los alumnos de esta generación se sienten cómodos con el uso de esta tecnología, por lo que no necesitarán un aprendizaje profundo de ésta. Al ser una herramienta de respuesta en tiempo real, contribuye a aumentar la participación y la atención de los alumnos en clase, haciéndolos partícipes del proceso de enseñanza y comprobando su progreso personal en la asignatura, lo cual es un incentivo positivo. Como alumno, el hecho de ser partícipe en una clase ayuda a no desviar la atención y a mejorar la asimilación de los conocimientos, puesto que el mayor problema en la enseñanza, es la distracción.

Los proyectos en tecnologías móviles son muy interesantes puesto que permiten partir de una base conceptual, hasta un producto final, en el que paso a paso se pueden comprobar los avances y, asimismo, como la tecnología nunca para de avanzar, surgirán nuevas soluciones y mejoras para aportar, por lo que siempre existe un margen tecnológico para la mejora.

Objetivos

La finalidad del proyecto consiste en construir una solución de software capaz de integrarse con los sistemas creados en 2015 de realización de test presenciales utilizando tecnologías móviles.

Es importante resaltar la importancia de que el test debe ser realizado presencialmente para que la calificación sea válida.

La tecnología seleccionada para este proyecto será iOS puesto que es un requisito del proyecto.

La arquitectura del sistema contará con una parte servidor, desde la cual el profesor podrá importar los formularios de preguntas, previamente realizados en Aula Global, y recibir las respuestas de los test de los alumnos, y una aplicación cliente, instalada en los Smartphone de los alumnos en los que se responderán las preguntas.

Estos test serán configurables con múltiples opciones al alcance de los profesores, tales como incorporar contraseñas, preguntas mono-respuesta, preguntas multi-respuesta, temporizador de respuesta, etc.

Otra cuestión a tener en cuenta es la interacción con el usuario o feedback, durante la realización del test o al finalizar el mismo. De este modo el alumno puede conocer el número de preguntas restantes, el porcentaje de acierto acumulado durante la prueba, la nota final, etc. En el lado del profesor, podrá comprobar el número de alumnos conectados, porcentaje de acierto por pregunta, tiempo medio de respuesta, etc.

Por lo tanto, los objetivos principales del proyecto son los que se enumeran a continuación:

- Creación de una aplicación en tecnología iOS, para responder preguntas tipo test.
- Integración de la aplicación con el servidor de preguntas disponible.

Planificación

Disponer de una buena planificación hace que el desarrollo de cualquier proyecto siempre es sinónimo de éxito, siendo de vital importancia a la hora de realizar aproximaciones o calcular presupuestos.

Tras cada etapa del proyecto se realizará un seguimiento de control para comprobar las posibles desviaciones existentes respecto al calendario fijado.

Para facilitar el manejo de ficheros, se dispondrá de un repositorio y se utilizará un control de versiones de los productos de software.

La planificación del proyecto viene marcada por la fecha de entrega del mismo, la cual se ha estimado a día 26 de septiembre de 2017. Se realizarán en paralelo el desarrollo y la documentación.

Al ser un proyecto cerrado de 300 horas de desarrollo¹ se ha podido realizar una planificación de cada parte importante del proyecto de la siguiente manera:

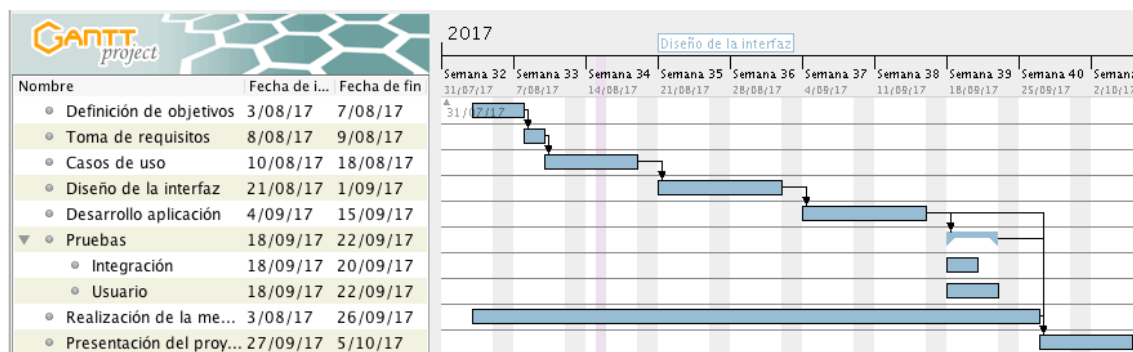


Ilustración 1 - Planificación del proyecto (GANTT)

¹ Cálculo en base a 25 horas por ETCS

Empezando el proyecto el día 2 de agosto y finalizando en la fecha prevista, 26 de septiembre.

Se han paralelizado todas aquellas tareas que no tienen dependencia fuerte con una anterior.

Estructura del documento

El documento queda formado por 8 apartados, incluyendo desarrollo del proyecto, bibliografía y anexos de la siguiente forma:

- Introducción: Introducción, contexto y motivación del proyecto.
- Estado de la cuestión: Analizar qué sistemas o soluciones existen para resolver el problema propuesto.
- Análisis, diseño, implementación e implantación: Metodologías utilizadas, ciclo de vida, requisitos, etc.
- Evaluación: Sistemas de evaluación, pruebas de usuario, etc.
- Planificación y presupuesto: Seguimiento de la planificación y análisis de costes.
- Conclusiones: Impresiones extraídas de la realización del proyecto.
- Anexo I: Manual de usuario: Pequeño manual de familiarización con la interfaz y uso de la aplicación por parte del alumno.
- Bibliografía y referencias: Inclusión de la fuente de los materiales consultados.

Estado del arte

En este apartado se realizará una introducción de las tecnologías actuales y que afectan de manera directa a la toma de decisiones en la realización del proyecto.

Al ser un requisito expreso del cliente, la realización del proyecto en la tecnología propietaria de Apple iOS, se realizará una comparación con el resto de tecnologías móviles del mercado y al ofrecer iOS dos posibles versiones, también se discutirá cual será la versión más conveniente.

Gracias a este estudio, se puede tomar una decisión acertada a la hora de comenzar el desarrollo del proyecto.

A parte del sistema operativo elegido para el desarrollo, se deben tener en cuenta elementos tales como los protocolos de comunicación, abanico de dispositivos objetivo, las alternativas a la solución, etc.

Sistema operativo

Como se ha comentado en puntos anteriores, la cuota actual de mercado en sistemas operativos para Smartphones, tiene tres grandes productos Android, Windows 10 Mobile e iOS. A continuación, se muestra una tabla comparativa de los tres sistemas operativos.

Android	iOS	Windows 10 Mobile
Compañía		
Open Handset Alliance	Apple Inc.	Microsoft
Última versión		
8.0 Oreo	11.0	10.0. 15252
Licencia		
Libre y de código abierto, pero por lo general se incluye con aplicaciones y drivers propietarios.	Propietaria excepto para componentes de código abierto.	Propietaria
Programado en		
C, C ++, Java.	C, C ++, Objective-C, Swift.	.NET C#, VB.NET, Silverlight, native C/C++, WinRTP (XMLA), DirectX
Tienda oficial de aplicaciones		
Google Play	App Store	Windows Store
Coste de desarrollo para el OS móvil		
Gratis	Gratis con Xcode 8	Gratis

USO DE TELÉFONOS IOS PARA CONTESTAR PREGUNTAS EN CLASES PRESENCIALES

Coste para publicar aplicación en la tienda oficial		
US\$25 una vez por individuo	US\$99 al año	US\$19, una vez por un individuo; y \$99 para una cuenta de la compañía
Soporte de impresoras		
4.4+ usando Google Cloud Print pero no a través de USB	Sí (AirPrint)	10+
Motor de navegador web por defecto		
Blink	WebKit	Trident (EdgeHTML después de la versión 10)
Navegadores web disponibles		
Chrome para Android, Opera, Firefox	Safari, Chrome para iOS, Opera Mini, Firefox	Internet Explorer, Opera Mini, UC Browser, Microsoft Edge
Motor de búsqueda de los navegadores		
Muchos (entre ellos Google)	Bing, Google, Yahoo! Search, DuckDuckGo	Muchos (entre ellos Bing)
Voz sobre IP		
Protocolo SIP o software de terceros	FaceTime y software de terceros	8+ Skype
Software de pago con tecnología NFC		
Disponible en cualquier dispositivo compatible con el hardware. Android Pay para pagos NFC disponible en Play Store.	8+: iPhone 6/6 Plus vía Apple Pay	8+
USB On-The-Go		
3.1 +	No	10+
Reproducción de audio		
AAC LC/LTP 3GPP, HE-AACv1 (AAC+), HE-AACv2 (AAC+ mejorado) AMR-NB, AMR-WB, MP3, MIDI (tipo 0 y 1, versiones DLS 1 y 2), Ogg Vorbis, PCM/WAVE, FLAC, WAVE, Opus	AAC, AAC protegido (del iTunes Store), HE-AAC, MP3, MP3 VBR, Audible (formatos 2, 3, 4, Audible Enhanced Audio, AAX y AAX+), Apple Lossless, AIFF, WAV	MP3, WMA Std 9.2, WMA Pro, FLAC, AMR-NB, AAC-LC, AAC+, eAAC+
Reproducción de vídeo		
H.263, H.264 (hasta Baseline Profile), H.265 HEVC, MPEG-4 SP, DivX, XviD, VP8, VP9	H.264 (hasta High Profile), MPEG-4, M-JPEG	H.263, H.264, WMV, MPEG4, MPEG4 @ HD 720p 30fps, DivX, XviD
Teclado Bluetooth		
2.3+, en versiones anteriores a través de software de terceros	Sí	8.1u2+
Teclado USB		
3.1+	Con el kit de conexión de cámara	10+ algunos dispositivos
Cliente SSH		
Sí	Sí	Sí

USO DE TELÉFONOS IOS PARA CONTESTAR PREGUNTAS EN CLASES PRESENCIALES

VPN		
Sí	Sí	8.1+
Reconocimiento de voz		
Sí	5+ (Siri)	8.1+ Microsoft Cortana

Tabla 1 - Comparativa sistemas operativos móviles

Una vez conocidas las diferencias, es importante conocer el porcentaje de cuota de mercado que ocupa cada uno actualmente dentro de los EU5(Alemania, Francia, Italia, España y Gran Bretaña):

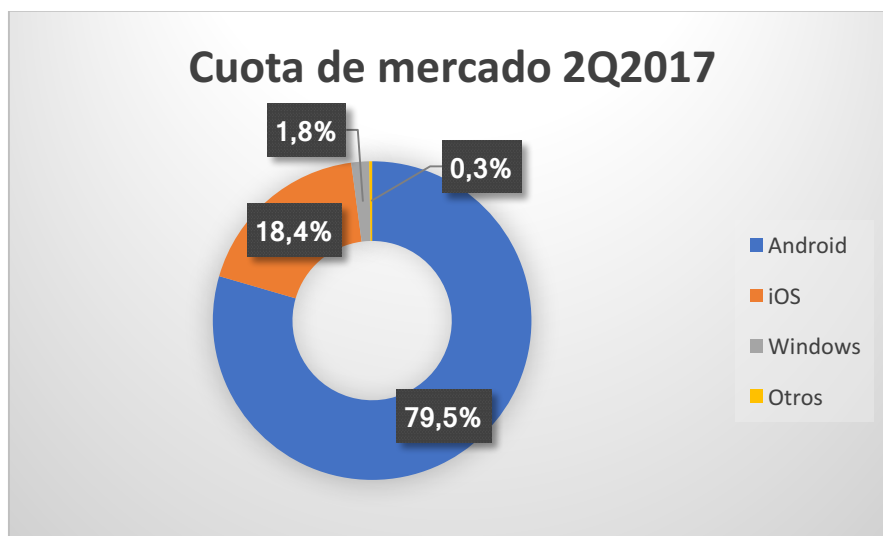
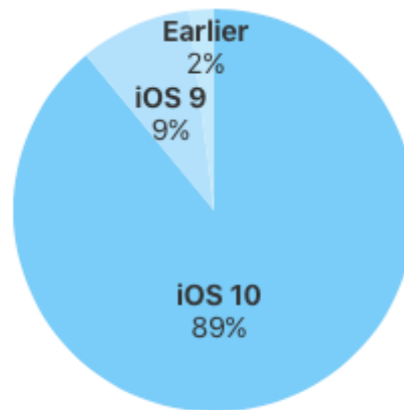


Ilustración 2 - Cuota mercado móvil 2Q17

Mientras que la segregación de versiones del sistema operativo es la siguiente:

89% of devices are using iOS 10.



As measured by the App Store on September 6, 2017.

Ilustración 3 - Adopción iOS por versión

En iOS 9 quedan un total del 9% de dispositivos compatibles, mientras que el 2% restante son versiones aún más antiguas como iOS 8 o iOS 7. Estos datos los recoge Apple analizando qué versión tiene cada dispositivo que descarga apps de la App Store. Lo que muestra que los usuarios de este sistema operativo mantienen sus dispositivos actualizados.

La elección del sistema operativo a utilizar viene dada por un requisito del cliente y será iOS. A continuación, se realizará un pequeño estudio sobre la elección del sistema operativo y los beneficios de este.

iOS

iOS es un sistema operativo para dispositivos móviles creado y desarrollado por Apple Inc.

Se trata de software propietario, especialmente diseñado para alguno de los dispositivos de la compañía, en concreto iPhone, iPad y iPod Touch. El resto de dispositivos de la compañía dispone de software específico para cada dispositivo restante como el Apple Watch con watchOS, Apple TV con tvOS o Mac con macOS.

iOS fue presentado durante la Macworld Conference & Expo, en el año 2007. En esta conferencia se presentaron dos novedades al mundo, el primer iPhone y la primera versión de su sistema operativo iPhone OS, basado en el sistema de escritorio de los ordenadores Apple, OS X, que a su vez está desarrollado en base Unix. La intención era portar un sistema operativo de escritorio a un dispositivo móvil como el iPod. Esto produjo una competición entre los equipos encargados del desarrollo de este sistema operativo, por un lado, el equipo de desarrollo del iPod y el de Scott Forstall, el cual se llevaría la victoria con la consecución del iPhone OS.

No fue hasta 2010 y la versión 4.0 el cambio de nomenclatura en el nombre del sistema operativo, pasando así a llamarse iOS. Este sistema operativo tiene actualizaciones anuales, y de manera periódica se publican parches con corrección de errores y mejoras. La versión más actual de iOS es la 11 (esta versión fue lanzada de manera oficial el 12 de septiembre de 2017, por lo que aún no hay datos comparativos).

Las características por las que iOS es famoso son las siguientes:

- Gran usabilidad y bonito diseño de una interfaz gráfica controlada por gestos táctiles.
- Migración de la experiencia de escritorio a un sistema móvil.
- Uso de aplicaciones similares a las que se pueden encontrar en un sistema de escritorio.

En sus inicios las aplicaciones externas no estaban soportadas, solo eran soportadas aplicaciones basadas en web. Gracias a la gran aceptación cambiaron este modelo y liberaron el primer SDK para desarrollo de aplicaciones nativas. Estas aplicaciones están basadas en el lenguaje de programación Objective-C. Desde entonces, su mercado de aplicaciones creció exponencialmente hasta convertirse en uno de los principales referentes como contenedor de apps de carácter comercial.

En 2014 Apple presentó Swift, un lenguaje de programación multiparadigma enfocado al desarrollo de aplicaciones para iOS y macOS. Puede integrarse con cualquier biblioteca programada en Objective-C y capaz de llamar a funciones de C. Está especialmente diseñado para integrarse con los Frameworks Cocoa y Cocoa Touch. En

2015 Swift pasó a ser de código abierto. La última versión de este lenguaje es la cuarta versión de este lenguaje de programación.

Xcode

A la hora de comenzar a desarrollar en cualquier lenguaje de programación la tarea más importante es elegir el IDE (Entorno de desarrollo integrado).

En este caso, para poder programar en iOS, es necesario disponer del IDE Xcode, disponible en el mercado de aplicaciones de manera gratuita de Apple. La primera versión de este entorno apareció en 2003 y actualmente se encuentra en la versión 8.

Xcode trabaja conjuntamente con Interface Builder, una herencia de NEXT, herramienta gráfica que facilita la creación de interfaces de usuario.

Sistemas similares

Una vez seleccionadas la tecnología a utilizar, la arquitectura de la solución y la conectividad, se deben estudiar las alternativas existentes en el mercado, que cubren la misma necesidad que el proyecto que se va a iniciar.

Socrative

Se trata de una herramienta educativa y gratuita utilizada para comprobar la adquisición de conocimientos por parte de los alumnos mediante rápidos test en tiempo real. Esta herramienta exige un registro para poder ser utilizada.

Formada por dos partes separadas, una para alumnos y otra para profesores, existente para las tres plataformas móviles analizadas y además una interfaz web.

El modo de uso consiste en que el profesor cree una sala virtual, crear los test y esperar a que los alumnos se conecten a esa sala, mediante un ID y una contraseña. Toda vez que los alumnos hayan finalizado su test, lo envían.

BlackBoard Learn

BlackBoard Learn es una herramienta LMS (sistemas de gestión del aprendizaje) similar a Moodle, donde se pueden compartir apuntes, exámenes, notas, realizar

llamamientos, etc. Por ello, diversas universidades alrededor del mundo lo han adoptado como herramienta corporativa.

Dentro de las posibilidades que ofrece, hay una aplicación llamada BlackBoard Mobile Learn, entre cuyas funciones se encuentra la de realizar test. El mecanismo es parecido al de todas las alternativas, el profesor crea un test a través de BlackBoard Learn y los alumnos pueden descargárselos en su móvil para realizarlos.

EduClick

La última herramienta a analizar es EduClick, este sistema amplía el abanico de posibilidades de dar una clase con presentaciones proyectadas. Si en el transcurso de la clase el profesor considera oportuno realizar un test, debe activar la función de inicio del periodo de respuesta, durante el cual, los alumnos deben contestar a la pregunta utilizando los dispositivos individuales.

Tabla comparativa

En la siguiente tabla comparativa se muestran los resultados tras analizar cada una de las posibles soluciones encontradas, enfrentándolas a lo que nuestra solución será capaz de hacer. Para ello, se tomarán como criterios de medida, su integración o no con Moodle, si es capaz de determinar si el alumno se encuentra físicamente en el aula, si los test son lanzados y respondidos en tiempo real y por último si el licenciamiento es gratuito o si necesita hardware adicional.

Los datos marcados en verde significan que cumplen con el requisito y los que están marcados en rojo significa que no es el resultado esperado.

	Moodle	Presencial	Tiempo real	Licenciamiento
Socrative	X	X	V	V
BlackBoard Learn	X	X	X	X
EduClick	V	V	V	V
Test_UC3M	V	V	V	X

Tabla 2 - Tabla comparativa soluciones similares

Como se puede observar en la tabla anterior, ninguna solución adopta el total de las necesidades del cliente, la opción que más se acerca es EduClick, que como se ha expuesto anteriormente, es la opción que se está utilizando actualmente en la universidad, pero al tener la necesidad de disponer de los mandos para realizar los test, con la consiguiente solicitud, puesto que hay unidades limitadas para el grueso del profesorado, hace que se necesite una solución adicional.

Como ninguna cubre todas las necesidades, se solicita la creación de una aplicación móvil, que englobe todos los requerimientos, siendo amigable, sencilla y útil.

Análisis, Diseño, Implementación e Implantación

Introducción

En el primer contacto con el cliente, se realiza una entrevista para poder definir los objetivos del proyecto, y poder así, facilitarles una estimación en tiempo y cuantía. Dicha estimación es un contrato vinculante por parte de ambos donde el cliente se compromete a abonar la cuantía acordada, siempre y cuando el proyecto finalice en el tiempo y forma acordado.

Para poder realizar la estimación de alto nivel, se necesita primero, conocer el objetivo del proyecto, en dicha reunión el cliente expone su necesidad:

“Necesitamos crear una aplicación móvil que nos permita evaluar a los alumnos mediante exámenes tipo test de manera presencial de forma rápida y sencilla. Puesto que el método actual no es del todo amigable, puesto que debo llevar unos mandos inalámbricos al aula, con los costes en batería y tiempo que esto conlleva. Por lo tanto, necesitamos una aplicación que pueda ser ejecutada en dispositivos móviles (tabletas y Smartphones) de los alumnos. Actualmente existe un servidor al que conectarse con la aplicación, mediante el cual se pueden crear test, configurarlos y exportarlos, de manera rápida y sencilla. Así pues, os pedimos, una aplicación móvil para terminales Apple cuyo sistema operativo es iOS, que se conecte mediante Wi-Fi a dicho servidor y sea capaz de mostrar los test, permitir responder a cada una de las preguntas, enviar los resultados al profesor y poder consultar un historial de resultados.”

Se continúa con la reunión, puesto que además existen una serie de restricciones a tener en cuenta:

No tendrá acceso a la plataforma Moodle de la Universidad Carlos III de Madrid, es decir, no se podrá realizar una integración total con la plataforma, si no que será el propio profesor el encargado de exportar los resultados a Moodle.

- La aplicación debe ser desarrollada en el sistema operativo iOS.
- El alumno debe estar presente en el aula a la hora de realizar el test.

Sin más, finalizó la reunión, y tras el análisis oportuno, se llega a la conclusión de que el problema es abordable y se puede lograr la consecución de todos los requisitos.

Análisis

En este apartado se llevará a cabo un análisis exhaustivo del sistema y de cada componente que formará parte de él.

Casos de uso

A continuación, se muestran los resultados de analizar el producto, se comenzará por el diagrama de casos de uso. Existirán dos actores:

- Profesor: Usuario de la aplicación de escritorio.
- Alumno: Usuario de la aplicación móvil.

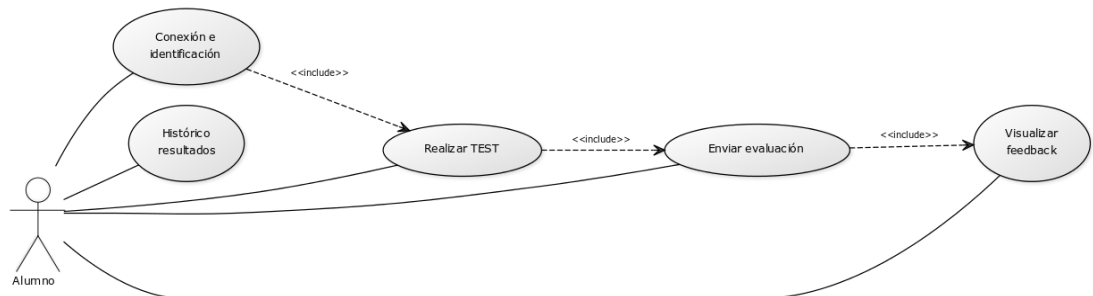


Ilustración 4 - Diagrama Casos de Uso Alumno

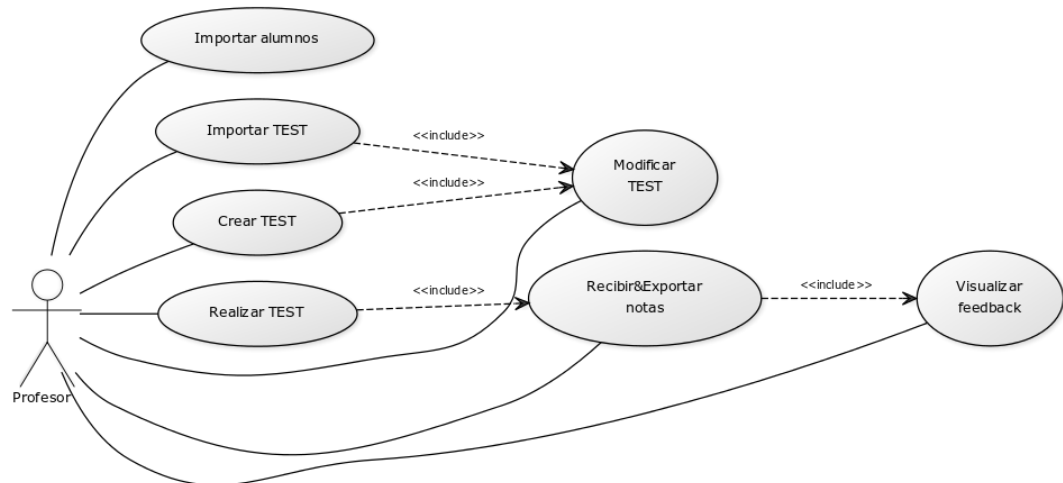


Ilustración 5 - Diagrama Casos de Uso Profesor

Los casos de uso del actor “Profesor” no serán analizados ni implementados en este proyecto, puesto que ya fueron desarrollados en los otros dos proyectos de los sistemas Android y Windows 10 Mobile, pero para tener una visión más completa del total del proyecto, es necesario explicar cada componente del mismo.

Descripción de los casos de uso

Se definirán los casos de uso utilizando como base la tabla expuesta a continuación:

Identificador	
Nombre	
Actor	
Objetivo	
Precondiciones	
Eventos	
Postcondiciones	

Tabla 3 - Plantilla de Casos de Uso

A continuación, se explican cada uno de los campos de la tabla a rellenar:

Identificador: valor con el que se hará referencia al caso de uso. Su nomenclatura será “CUXX”, donde “CU” hace referencia al objeto, en este caso, caso de uso y “XX” a un valor numérico secuencial.

- **Nombre:** nombre asociado al diagrama.
- **Actor:** agente que interactúa en el caso de uso.
- **Objetivo:** descripción del caso de uso.
- **Precondiciones:** condiciones iniciales que deben cumplirse para poder lanzar este caso de uso.
- **Eventos:** acciones realizadas en el caso de uso.
- **Postcondiciones:** estado del sistema después del caso de uso.

Tras esta pequeña descripción se muestran todos los casos de uso de manera pormenorizada:

CU01	
Nombre	Importar TEST
Actor	Profesor
Objetivo	Posibilidad de importar TEST en formato admitido
Precondiciones	<ul style="list-style-type: none"> - Fichero TEST en formato admitido - Aplicación iniciada
Eventos	<ol style="list-style-type: none"> 1. Ejecutar el programa de escritorio 2. Seleccionar la opción “Importar TEST” 3. Seleccionar el fichero de test correspondiente 4. Hacer clic en “Abrir”
Postcondiciones	El test se importa y se visualiza en la lista de test disponibles

Tabla 4 - CU01 – Importar TEST

CU02	
Nombre	Importar alumno
Actor	Profesor
Objetivo	Posibilidad de importar el listado de alumnos en el formato admitido
Precondiciones	<ul style="list-style-type: none"> - Lista de alumnos en formato admitido - Aplicación iniciada
Eventos	<ol style="list-style-type: none"> 1. Ejecutar el programa de escritorio 2. Seleccionar la opción “Importar Alumnos” 3. Seleccionar el fichero de alumnos correspondiente 4. Hacer clic en “Abrir”
Postcondiciones	La lista de alumnos se importa y se visualiza en la lista de alumnos

Tabla 5 - CU02 – Importar alumno

CU03	
Nombre	Crear TEST
Actor	Profesor
Objetivo	Crear un nuevo test
Precondiciones	Aplicación iniciada
Eventos	<ol style="list-style-type: none"> 1. Ejecutar el programa de escritorio 2. Rellenar el formulario de creación 3. Crear y agrupar las preguntas
Postcondiciones	<ul style="list-style-type: none"> - El profesor puede visualizar el diseño final del cuestionario - Se almacenan los datos en el fichero creado para tal fin

Tabla 6 - CU03 – Crear TEST

CU04	
Nombre	Modificar TEST
Actor	Profesor
Objetivo	Modificar TEST ya creados
Precondiciones	<ul style="list-style-type: none"> - Aplicación iniciada - TEST importado
Eventos	<ol style="list-style-type: none"> 1. Ejecutar el programa de escritorio 2. Importar TEST 3. Seleccionar “Modificar Test” 4. Seleccionar la pregunta que se desea modificar 5. Realizar la edición 6. Volver al paso 4 tantas veces como sea necesario
Postcondiciones	<ul style="list-style-type: none"> - El profesor puede visualizar el diseño final del cuestionario - Se almacenan los datos en el fichero creado para tal fin

Tabla 7 - CU04 – Modificar TEST

CU05	
Nombre	Realizar TEST
Actor	Profesor
Objetivo	Lanzar el test a los alumnos para iniciarlo
Precondiciones	<ul style="list-style-type: none"> - Aplicación iniciada - Test importado
Eventos	<ol style="list-style-type: none"> 1. Ejecutar el programa de escritorio 2. Importar TEST 3. Hacer clic en el botón “Realizar TEST” 4. Seleccionar las características del TEST 5. Esperar a que los alumnos se conecten 6. Hacer clic en el botón “Comenzar TEST”
Postcondiciones	En la aplicación de escritorio aparecerá toda la información relativa a los resultados del test

Tabla 8 - CU05 – Realizar TEST

CU06	
Nombre	Recibir&Exportar notas
Actor	Profesor
Objetivo	Poder visualizar y exportar los resultados del test
Precondiciones	<ul style="list-style-type: none"> - Aplicación iniciada - Test importado - Test realizado - Test finalizado
Eventos	<ol style="list-style-type: none"> 1. Ejecutar el programa de escritorio 2. Importar TEST 3. Realizar TEST 4. Esperar resultados 5. Hacer clic en el botón “Exportar calificaciones” 6. Seleccionar el fichero destino
Postcondiciones	Fichero con los resultados de la realización del test en la ruta indicada

Tabla 9 - CU06 – Recibir&Exportar notas

CU07	
Nombre	Visualizar feedback
Actor	Profesor
Objetivo	Poder visualizar las estadísticas del TEST
Precondiciones	<ul style="list-style-type: none"> - Aplicación iniciada - Test importado - Test realizado - Test finalizado
Eventos	<ol style="list-style-type: none"> 1. Ejecutar el programa de escritorio 2. Importar TEST 3. Realizar TEST 4. Esperar resultados 5. Hacer clic sobre el botón “Feedback”
Postcondiciones	Se muestra la información

Tabla 10 - CU07 – Visualizar feedback

CU08	
Nombre	Conexión e identificación
Actor	Alumno
Objetivo	Poder conectarse al servidor
Precondiciones	<ul style="list-style-type: none"> - Aplicación servidor iniciada - Aplicación cliente iniciada
Eventos	<ol style="list-style-type: none"> 1. Rellenar el formulario de conexión 2. Pulsar el botón “Conectar”
Postcondiciones	<ul style="list-style-type: none"> - Aparecerá un mensaje con el resultado (Éxito / Error) - Aparecerá el NIA en la pantalla del servidor

Tabla 11 - CU08 – Conexión e identificación

CU09	
Nombre	Realizar TEST
Actor	Alumno
Objetivo	Realizar el test recibido en la aplicación cliente
Precondiciones	<ul style="list-style-type: none"> - Aplicación iniciada - Conexión exitosa
Eventos	<ol style="list-style-type: none"> 1. Seleccionar la opción deseada para responder a la pregunta 2. Pulsar siguiente 3. Realizar el paso 1 hasta que aparezca el botón “Finalizar”
Postcondiciones	Se mostrará un mensaje de confirmación de envío y finalización del TEST

Tabla 12 - CU09 – Realizar TEST

CU10	
Nombre	Enviar evaluación
Actor	Alumno
Objetivo	Enviar las respuestas elegidas al servidor
Precondiciones	<ul style="list-style-type: none"> - Aplicación iniciada - Conexión exitosa - Test finalizado
Eventos	1. Confirmar la finalización y envío del test
Postcondiciones	<ul style="list-style-type: none"> - Se enviarán los datos del test al servidor - Se notificará al usuario del resultado

Tabla 13 - CU10 – Enviar evaluación

CU11	
Nombre	Visualizar feedback
Actor	Alumno
Objetivo	Poder visualizar las estadísticas del TEST
Precondiciones	<ul style="list-style-type: none"> - Aplicación iniciada - Conexión exitosa - Test finalizado y enviado
Eventos	N/A
Postcondiciones	En caso de que esté habilitado, se mostrará la calificación del alumno al finalizar el test

Tabla 14 - CU11 – Visualizar feedback

CU12	
Nombre	Histórico resultados
Actor	Alumnos
Objetivo	Poder visualizar las calificaciones obtenidas en los distintos test
Precondiciones	Aplicación iniciada
Eventos	1. Pulsar sobre el botón “Histórico de calificaciones”
Postcondiciones	Se muestra un listado con las distintas calificaciones para cada uno de los test realizados

Tabla 15 - CU12 – Histórico resultados

Especificación de los requisitos

La parte más importante de cualquier proyecto de software son los requisitos, puesto que componen las peticiones y necesidades del cliente y las explican de manera detallada.

Descripción de los requisitos

A continuación, se muestra la tabla usada como plantilla de los requisitos:

Identificador		
Nombre:		Fuente: <input type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:		
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 16 - Plantilla de requisitos

Para facilitar el entendimiento de cada campo de la tabla, se explicarán detalladamente cada uno de ellos:

- **Identificador:** valor con el que se hará referencia al requisito con la siguiente nomenclatura: “RSXXX-DDD”:
 - RS: significa requisito de software
 - XXX: indica el tipo de requisito.
 - FNC: Requisitos funcionales
 - NFN: Requisitos no funcionales
 - RND: Requisitos de rendimiento
 - ITF: Requisitos de interfaz
 - RSC: Requisitos de recursos
 - CMP: Requisitos de comprobación
 - ACP: Requisitos de aceptación de las pruebas
 - SGD: Requisitos de seguridad
 - QTY: Requisitos de calidad
 - DDD: Valor numérico secuencial.
- **Nombre:** Identificación unívoca del requisito, ha de ser descriptiva.
- **Fuente:** Aclara el origen del requisito, puede tomar dos valores:
 - Cliente: Requisito impuesto por el cliente
 - Desarrollador: Requisito impuesto por el equipo de desarrollo
- **Descripción:** Breve resumen del contenido del requisito.

- **Prioridad:** Valores de importancia del cumplimiento del requisito. Marcan un orden en la consecución del requisito, puede tomar los valores:
 - Alta: carácter primordial
 - Media: carácter normal
 - Bajo: carácter de baja importancia
- **Necesidad:** Indica si el requisito debe incluirse en el producto. Puede tener los siguientes valores:
 - Esencial: Debe introducirse en el producto final
 - Deseable: Debe introducirse en el producto final, siempre y cuando no cree colisiones.
 - Opcional: El requisito puede no introducirse en el sistema si así se considera oportuno
- **Estabilidad:** Posibilidad de que el requisito sea modificable a lo largo del desarrollo del producto. Puede tener los siguientes valores:
 - Alta: No puede modificarse el requisito
 - Medio: El requisito puede ser modificado de manera frecuente
 - Bajo: El requisito puede ser modificado con una frecuencia alta
- **Verificabilidad:** Indica si se puede comprobar el requisito en el producto. Puede tomar los siguientes valores:
 - Alta: Comprobación sencilla
 - Media: Comprobación costosa, aunque no imposible
 - Baja: Comprobación imposible

Requisitos funcionales

RSFNC-001		
Nombre:	Login	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	El alumno debe identificarse para acceder a los test	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 17 - RSFNC-001

RSFNC-002		
Nombre:	Obtener TEST	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	La aplicación recibirá los test cuando el servidor los envíe	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 18 - RSFNC-002

RSFNC-003		
Nombre:	Visualizar TEST	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Una vez enviado el test por el servidor, debe ser visualizado en la aplicación cliente.	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 19 - RSFNC-003

RSFNC-004		
Nombre:	Acceso a test sin contraseña	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Si en la aplicación servidor se ha definido el acceso sin contraseña, el test debe ser mostrado	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 20 - RSFNC-004

RSFNC-005		
Nombre:	Acceso a test con contraseña	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Si en la aplicación servidor se ha configurado un test con contraseña, el usuario debe introducirla en la aplicación cliente.	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 21 - RSFNC-005

RSFNC-006		
Nombre:	Contraseña	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	En el caso de producirse el RSFNC-005 debe mostrarse la interfaz de inclusión de contraseña	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 22 - RSFNC-006

RSFNC-007		
Nombre:	Visualización preguntas	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	La aplicación cliente debe mostrar las preguntas creadas por la aplicación servidor	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 23 - RSFNC-007

RSFNC-008		
Nombre:	Visualización respuestas	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	La aplicación cliente debe mostrar las posibles respuestas a cada pregunta	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 24 - RSFNC-008

RSFNC-009		
Nombre:	Envío respuestas	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	La aplicación cliente debe tener un botón para el envío de las respuestas	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 25 - RSFNC-009

RSFNC-010		
Nombre:	Control en la respuesta	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	La aplicación cliente debe determinar el tipo de respuesta, a partir de la configuración dada por el servidor	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 26 - RSFNC-010

RSFNC-011		
Nombre:	Feedback	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	Tras finalizar el test, y si la configuración de la aplicación servidor así lo permite, se mostrará información relacionada con el proceso de evaluación, tal como, porcentaje de acierto, distribución de la nota, etc.	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 27 - RSFNC-011

RSFNC-012		
Nombre:	Conexión Wi-Fi	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	La aplicación cliente se conectará al servidor mediante Wi-Fi	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 28 - RSFNC-012

RSFNC-013		
Nombre:	Histórico	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	La aplicación cliente podrá mostrar el histórico de calificaciones en los test que contengan feedback personal	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 29 - RSFNC-013

RSFNC-014		
Nombre:	Recepción confirmación	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Una vez finalizado el proceso de evaluación la aplicación cliente recibirá una confirmación.	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 30 - RSFNC-014

RSFNC-015		
Nombre:	Interrumpir TEST	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	La aplicación servidor podrá interrumpir un test, el cliente notificará el cierre del test.	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 31 - RSFNC-015

Requisitos no funcionales

RSNFN-001		
Nombre:	Compatibilidad iOS	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	La aplicación será compatible con dispositivos con versiones iOS 10.2 y superiores	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 32 - RSNFN-001

RSNFN-002		
Nombre:	Normas de estilo	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Se respetarán las normas de estilo definidas por la documentación oficial de iOS	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 33 - RSNFN-002

RSNFN-003		
Nombre:	Resolución de pantalla	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Las resoluciones serán todas las de los terminales Apple	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 34 - RSNFN-003

RSNFN-004		
Nombre:	Documentación	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Se seguirán las normas impuestas por Métrica V3	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 35 - RSNFN-004

Requisitos de rendimiento

RSRND-001		
Nombre:	Tiempo de respuesta	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	El tiempo de respuesta del sistema ha de ser inferior a los dos segundos	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	

Tabla 36 - RSRND-001

RSRND-002		
Nombre:	Acceso a información	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	El acceso a la información histórica de los test no deberá ser superior al segundo	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	

Tabla 37 - RSRND-002

RSRND-003		
Nombre:	Notificaciones	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Las notificaciones del sistema deben ser mostradas en menos de un segundo	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 38 - RSRND-003

Requisitos de interfaz

RSITF-001		
Nombre:	Vista HOME	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	Vista inicial, contendrá: Botón → Realizar TEST Botón → Histórico TEST	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 39 - RSITF-001

RSITF-002		
Nombre:	Vista login	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	Vista con los datos para realizar la conexión con el servidor, contendrá: Input → IP Input → PORT Input → NIA Botón → Conectar	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 40 - RSITF-002

RSITF-003		
Nombre:	Vista respuestas	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	Vista donde se mostrará la interfaz de las respuestas	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 41 - RSITF-003

RSITF-004		
Nombre:	Vista pregunta	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	Vista que contendrá tanto la pregunta, como la zona de respuestas	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 42 - RSITF-004

RSITF-005		
Nombre:	Vista tiempo	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Vista de respuesta que contendrá un contador recesivo de tiempo	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 43 - RSITF-005

RSITF-006		
Nombre:	Vista finalización	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Vista que contendrá el feedback del TEST en caso de estar habilitado	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 44 - RSITF-006

Requisitos de recursos

RSRSC-001		
Nombre:	Consumo de batería	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	El consumo de batería no será mayor al 2%	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 45 - RSRSC-001

RSRSC-002		
Nombre:	Datos histórico	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	Los datos se almacenarán en un fichero interno, no deberá ocupar más de 1 MB	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 46 - RSRSC-002

RSRSC-003		
Nombre:	Tamaño de la aplicación cliente	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	El tamaño final de la aplicación no deberá superar los 20MB	
Prioridad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 47 - RSRSC-003

RSRSC-004		
Nombre:	Conexión local	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	La aplicación cliente se conectará de manera local al servidor a través de un puerto definido por el profesor en la aplicación servidor, por defecto será el puerto TCP 80	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 48 - RSRSC-004

RSRSC-005		
Nombre:	Comunicación	Fuente: <input checked="" type="checkbox"/> Cliente <input type="checkbox"/> Desarrollador
Descripción:	Las comunicaciones entre la aplicación cliente y servidor serán mediante un socket TCP/IP	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 49 - RSRSC-005

Requisitos de comprobación

RSCMP-001		
Nombre:	Comprobación IP	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	El formato de ip válido será: XXX.XXX.XXX.XXX Dónde XXX es un número entre 0 y 255	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 50 - RSCMP-001

RSCMP-002		
Nombre:	Comprobación puerto	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Debe ser un número entre 0 y 65535	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 51 - RSCMP-002

RSCMP-003		
Nombre:	Comprobación NIA	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	El formato de NIA válido será: XXXXXXXXXX Dónde X es un dígito que puede valer entre 0 y 9	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 52 - RSCMP-003

RSCMP-004		
Nombre:	Éxito conexión	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Si la conexión se realiza correctamente debe mostrarse un mensaje	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 53 - RSCMP-004

RSCMP-005		
Nombre:	Respuesta vacía	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	En caso de abandonar una respuesta en blanco, debe confirmarse	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 54 - RSCMP-005

Requisitos de aceptación

RSACP-001		
Nombre:	Pruebas del sistema	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Debe superar las pruebas de sistema	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 55 - RSACP-001

RSACP-002		
Nombre:	Pruebas de usuario	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Debe superar las pruebas de usuario	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 56 - RSACP-002

RSACP-003		
Nombre:	Pruebas de integración	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Debe superar las pruebas de integración	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 57 - RSACP-003

RSACP-004		
Nombre:	Pruebas de aceptación	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Debe superar las pruebas de aceptación	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 58 - RSACP-004

RSACP-005		
Nombre:	Pruebas de regresión	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Debe superar las pruebas de regresión	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 59 - RSACP-005

RSACP-006		
Nombre:	Pruebas unitarias	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Debe superar las pruebas unitarias	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 60 - RSACP-006

Requisitos de seguridad

RSASGD-001		
Nombre:	Comunicación	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Se deben cifrar las conexiones y comunicaciones entre cliente y servidor	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input checked="" type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 61 - RSSGD-001

RSASGD-002		
Nombre:	LOPD	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	No se manejará información sensible del usuario ni será revelada a otros usuarios	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 62 - RSSGD-002

RSASGD-003		
Nombre:	Conexiones máximas	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Se limitará a 3 las conexiones desde una misma IP	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 63 - RSSGD-003

RSASGD-004		
Nombre:	Inactividad	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	En caso de inactividad durante 90 minutos se cerrará la conexión	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 64 - RSSGD-004

RSASGD-005		
Nombre:	Desconexión	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	En caso de realizarse una desconexión no esperada, deberá manejarse mediante el uso de excepciones controladas.	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 65 - RSSGD-005

Requisitos de calidad

RSAQTY-001		
Nombre:	Interfaz	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	La interfaz debe ser intuitiva y práctica	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 66 - RSQTY-001

RSAQTY-002		
Nombre:	Documentación	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	La documentación será entregada sin fallos gramaticales ni ortográficos, siguiendo el estándar establecido	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 67 - RSQTY-002

RSAQTY-003		
Nombre:	Fiabilidad	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	La fiabilidad del sistema estará por encima del 95%	
Prioridad: <input type="checkbox"/> Alta <input checked="" type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input checked="" type="checkbox"/> Opcional	
Estabilidad: <input type="checkbox"/> Alta <input type="checkbox"/> Media <input checked="" type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 68 - RSQTY-003

RSAQTY-004		
Nombre:	Pruebas	Fuente: <input type="checkbox"/> Cliente <input checked="" type="checkbox"/> Desarrollador
Descripción:	Todas las pruebas serán realizadas en el peor de los casos	
Prioridad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Necesidad: <input checked="" type="checkbox"/> Esencial <input type="checkbox"/> Deseable <input type="checkbox"/> Opcional	
Estabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	Verificabilidad: <input checked="" type="checkbox"/> Alta <input type="checkbox"/> Media <input type="checkbox"/> Baja	

Tabla 69 - RSQTY-004

Matriz de trazabilidad

En la siguiente tabla se muestra la trazabilidad entre los requisitos funcionales y los casos de uso del actor alumno:

	CU08	CU09	CU10	CU11	CU12
RSFNC-001	X				
RSFNC-002		X			
RSFNC-003		X			
RSFNC-004		X			
RSFNC-005		X			
RSFNC-006		X			
RSFNC-007		X			
RSFNC-008		X			
RSFNC-009			X		
RSFNC-010		X			
RSFNC-011				X	
RSFNC-012	X				
RSFNC-013					X
RSFNC-014		X			
RSFNC-015		X			

Tabla 70 - Trazabilidad entre requisitos funcionales y casos de uso

Diseño

Una vez terminada la fase de análisis, se debe realizar un diseño final de la aplicación usando todas las especificaciones que han sido detalladas en apartados anteriores. Se deberán definir la arquitectura y el entorno tecnológico que le va a dar soporte, así como los componentes tecnológicos involucrados.

Asimismo, se describirán de manera detallada los casos de uso reales, y el diagrama de clases. También, se recoge el diseño final de la interfaz.

Este proyecto se centra únicamente en la parte cliente, no obstante, es necesario conocer el funcionamiento global de la solución y cada uno de sus componentes.

Arquitectura del sistema

Al necesitar la solución integrarse con la arquitectura actual implementada, siendo esta cliente-servidor, será el servidor el que se encargue de la mayor carga de trabajo, puesto que será ejecutado en una máquina con mayores especificaciones que los clientes.

Por parte de los clientes, este sistema estará disponible para tres sistemas operativos móviles, Android, Windows 10 Mobile e iOS centrándose este proyecto en esta última tecnología. La elección de una arquitectura cliente-servidor maximiza la compatibilidad multiplataforma, puesto que las operaciones serán invocadas a través de un protocolo de comunicación neutro.



Ilustración 6 - Arquitectura del sistema

Por lo tanto, queda dividido en dos sistemas de desarrollo:

- Servidor: Aplicación para escritorio desarrollada en Java 1.8 ya finalizado y desplegado
- Cliente: Aplicación desarrollada en Swift 4.0 para dispositivos Apple iPhone y iPad con una versión de iOS 10.3 o superior.

Normas de diseño y construcción y estándares

Para una correcta construcción del sistema, deben definirse una serie de normas y estándares a cumplir, donde se definen, entre otros, el lenguaje de programación a utilizar, el idioma, nomenclatura de variables, etc.

- Lenguaje de programación: Se utilizará Swift 4.0 y en entorno de desarrollo Xcode.
- Idioma del sistema: En una primera instancia será español, pero a futuro se podrán incluir más idiomas, siendo utilizado el predefinido por el sistema.
- Nombrado de clases: La primera letra debe ser mayúscula y el resto minúsculas, si el nombre tuviese más de una palabra, se permite usar el separador guion bajo o la primera letra de cada palabra en mayúsculas.
- Nombrado de variables: deben estar escritas en minúsculas permitiendo el uso de guiones bajos y números. Han de ser descriptivos.
- Nombrado de constantes: Han de estar escritas en mayúsculas.
- Nombrado de vistas: Deben terminar por ViewController y han de ser descriptivas.
- Comentarios: En el caso de ser líneas simples se utilizará al principio de la misma “//”, en el caso de ser múltiples líneas deben estar encapsuladas por “/*” y “*/”.

Entorno tecnológico

Las necesidades tecnológicas asociadas a este proyecto son las descritas a continuación.

Como se ha descrito en apartados anteriores, se desarrollará una aplicación para dispositivos móviles de Apple, utilizando iOS, siendo compatible con su versión 10.3 y

versiones más modernas. El estándar para la comunicación punto a punto será el IEEE 802.11 o estándar Wi-Fi.

El motivo de esta elección es la compatibilidad entre ambas, así como la fiabilidad del sistema iOS y la consistencia del estándar Wi-Fi.

El entorno software utilizado para el desarrollo de la aplicación será el siguiente:

- macOS Sierra (10.12.6): Versión más actual del sistema operativo en el momento de realizar el desarrollo del proyecto.
- Xcode (8.2): Versión más actual del IDE.

Conectividad

- La conectividad será mediante conexión Wi-Fi, ya que es un requisito del proyecto. Esta conectividad permite controlar la presencia, puesto que es necesario que el cliente esté conectado a la red interna de la Universidad Carlos III de Madrid. Ya que todos los Smartphones actuales disponen de módulo Wi-Fi esto no será un impedimento. El protocolo TCP/IP ofrece una conexión segura y fiable.

Diseño de clases

A continuación, se definen los componentes del sistema y la relación entre ellos.

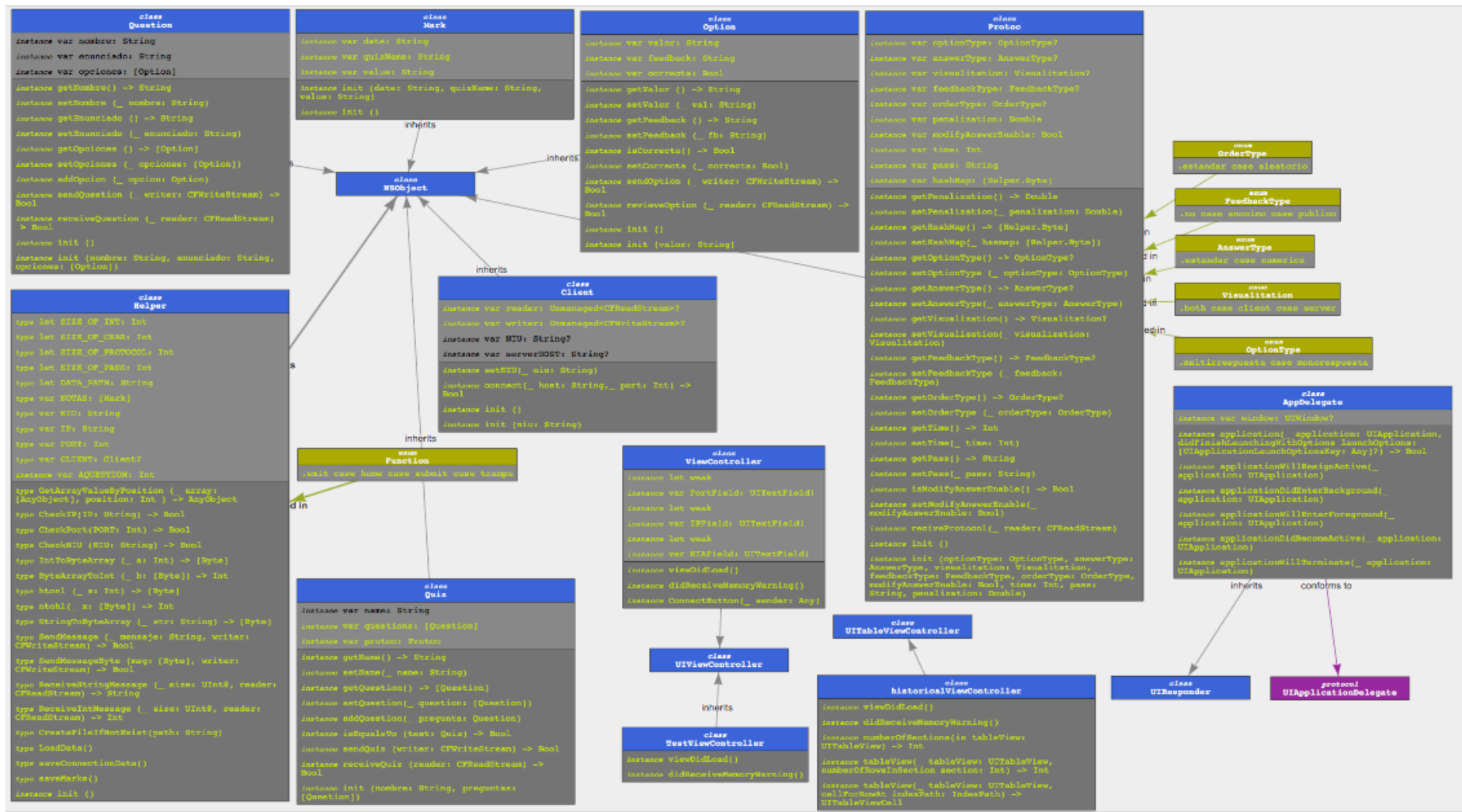


Ilustración 7 - Diagrama de clases

A continuación, se pasa a describir cada uno de los componentes de manera más detallada:

AppDelegate

```
class AppDelegate
instance var window: UIWindow?

instance application(_ application: UIApplication,
didFinishLaunchingWithOptions launchOptions:
[UIApplicationLaunchOptionsKey: Any]?) -> Bool

instance applicationWillResignActive(_
application: UIApplication)

instance applicationDidEnterBackground(_
application: UIApplication)

instance applicationWillEnterForeground(_
application: UIApplication)

instance applicationDidBecomeActive(_ application:
UIApplication)

instance applicationWillTerminate(_ application:
UIApplication)
```

Ilustración 8 - Clase AppDelegate

En esta clase se recogen los eventos del sistema lanzados al cargar las opciones de la aplicación, o cuando la aplicación es enviada al background. En este proyecto no se contemplan personalizaciones para esta clase.

Client

```
class Client
instance var reader: Unmanaged<CFReadStream>?
instance var writer: Unmanaged<CFWriteStream>?
instance var NIU: String?
instance var serverHOST: String?

instance setNIU(_ niu: String)

instance connect(_ host: String, _ port: Int) ->
Bool

instance init ()
instance init (niu: String)
```

Ilustración 9 - Clase Client

Es la representación lógica del usuario. En esta clase se almacena toda la información necesaria para mantener la comunicación con el servidor, véase, dirección IP, socket, canales de escritura y lectura. A su vez, almacena las respuestas para ser enviadas al servidor al finalizar el test.

Mark

```

class
Mark

instance var date: String
instance var quizName: String
instance var value: String

instance init (date: String, quizName: String,
value: String)
instance init ()

```

Ilustración 10 - Clase Mark

En esta clase se representa el tipo complejo de datos para la calificación de un usuario, relacionando a un alumno con la calificación obtenida en un determinado test.

Option

```

class
Option

instance var valor: String
instance var feedback: String
instance var correcta: Bool

instance getValor () -> String
instance setValor (_ val: String)
instance getFeedback () -> String
instance setFeedback (_ fb: String)
instance isCorrecta() -> Bool
instance setCorrecta (_ correcta: Bool)
instance sendOption (_ writer: CFWriteStream) ->
Bool
instance revieveOption (_ reader: CFReadStream) ->
Bool
instance init ()
instance init (valor: String)

```

Ilustración 11 - Clase Option

En esta clase se representan cada una de las opciones que componen una pregunta de un test. También implementa la comunicación para recibir y enviar toda la información que contiene.

Question

```
class
Question

instance var nombre: String
instance var enunciado: String
instance var opciones: [Option]

instance getNombre() -> String
instance setNombre (_ nombre: String)
instance getEnunciado () -> String
instance setEnunciado (_ enunciado: String)
instance getOpciones () -> [Option]
instance setOpciones (_ opciones: [Option])
instance addOpcion (_ opcion: Option)
instance sendQuestion (_ writer: CFWriteStream) ->
Bool
instance receiveQuestion (_ reader: CFReadStream)
> Bool
instance init ()
instance init (nombre: String, enunciado: String,
opciones: [Option])
```

Ilustración 12 - Clase Question

En esta clase se representan cada una de las preguntas que componen un test, contiene información sobre las opciones, el nombre de la pregunta, el enunciado, etc. También implementa la comunicación para recibir y enviar toda la información que contiene.

Protoc

```

class
Protoc

instance var optionType: OptionType?
instance var answerType: AnswerType?
instance var visualitation: Visualitation?
instance var feedbackType: FeedbackType?
instance var orderType: OrderType?
instance var penalization: Double
instance var modifyAnswerEnable: Bool
instance var time: Int
instance var pass: String
instance var hashMap: [Helper.Byte]

instance getPenalization() -> Double
instance setPenalization(_ penalization: Double)
instance getHashMap() -> [Helper.Byte]
instance setHashMap(_ hasmap: [Helper.Byte])
instance getOptionType() -> OptionType?
instance setOptionType (_ optionType: OptionType)
instance getAnswerType() -> AnswerType?
instance setAnswerType(_ answerType: AnswerType)
instance getVisualization() -> Visualitation?
instance setVisualization(_ visualization:
Visualitation)
instance getFeedbackType() -> FeedbackType?
instance setFeedbackType (_ feedback:
FeedbackType)
instance getOrderType() -> OrderType?
instance setOrderType (_ orderType: OrderType)
instance getTime() -> Int
instance setTime(_ time: Int)
instance getPass() -> String
instance setPass(_ pass: String)
instance isModifyAnswerEnable() -> Bool
instance setModifyAnswerEnable(_
modifyAnswerEnable: Bool)
instance receiveProtocol(_ reader: CFReadStream)
instance init ()

instance init (optionType: OptionType, answerType:
AnswerType, visualitation: Visualitation,
feedbackType: FeedbackType, orderType: OrderType,
modifyAnswerEnable: Bool, time: Int, pass:
String, penalization: Double)

```

Ilustración 13 - Clase Protoc

En esta clase se representa la configuración determinada para un test. Es la encargada de recoger la configuración enviada por el servidor, así como generar y decodificar el hash enviado al servidor con la configuración del test.

Quiz

```
class
Quiz

instance var name: String
instance var questions: [Question]
instance var protoc: Protoc

instance getName() -> String
instance setName(_ name: String)
instance getQuestion() -> [Question]
instance setQuestion(_ question: [Question])
instance addQuestion(_ pregunta: Question)
instance isEqualTo (test: Quiz) -> Bool
instance sendQuiz (writer: CFWriteStream) -> Bool
instance receiveQuiz (reader: CFReadStream) ->
Bool
instance init (nombre: String, preguntas:
[Question])
```

Ilustración 14 - Clase Quiz

En esta clase se representa la totalidad del examen tipo test. En ella están contenidas cada una de las preguntas y opciones que componen el test, y el nombre del mismo. También implementa la comunicación para enviar y recibir toda la información que contiene.

Helper

```

class
  Helper

type let SIZE_OF_INT: Int
type let SIZE_OF_CHAR: Int
type let SIZE_OF_PROTOCOL: Int
type let SIZE_OF_PASS: Int
type let DATA_PATH: String
type var NOTAS: [Mark]
type var NIU: String
type var IP: String
type var PORT: Int
type var CLIENT: Client?
instance var AQUESTION: Int

type GetArrayValueByPosition (_ array:
  [AnyObject], position: Int ) -> AnyObject
type CheckIP(IP: String) -> Bool
type CheckPort(PORT: Int) -> Bool
type CheckNIU (NIU: String) -> Bool
type IntToByteArray (_ a: Int) -> [Byte]
type ByteArrayToInt (_ b: [Byte]) -> Int
type htonl (_ x: Int) -> [Byte]
type etohl(_ x: [Byte]) -> Int
type StringToByteArray (_ str: String) -> [Byte]
type SendMessage (_ mensaje: String, writer:
  CFWriteStream) -> Bool
type SendMessageByte (msg: [Byte], writer:
  CFWriteStream) -> Bool
type ReceiveStringMessage (_ size: UInt8, reader:
  CFReadStream) -> String
type ReceiveIntMessage (_ size: UInt8, reader:
  CFReadStream) -> Int
type CreateFileIfNotExist(path: String)
type LoadData()
type saveConnectionData()
type saveMarks()
instance init ()

```

Ilustración 15 - Clase Helper

Clase de apoyo con todo tipo de métodos que no pueden ser categorizados en otras clases. En esta clase se han implementado métodos para ayudar a realizar conversiones, enviar mensajes por los canales de escritura, etc.

Protocolo de comunicaciones

Para entender perfectamente el funcionamiento y la conexión existente entre la aplicación cliente y la aplicación servidor, es de vital importancia conocer el protocolo de comunicación que ha sido creado para tal fin.

Para evitar la pérdida de información o la corrupción de los datos debido a que el almacenamiento de los datos depende tanto del software como del hardware, los bytes que componen estos datos pueden ser guardados de distintas formas. Para evitar esta interpretación errónea de los datos se optó por utilizar el llamado marshalling o

aplanamiento de datos. Con esta técnica se consigue evitar el mantenimiento entre máquinas que utilizan Big Endian o Little Endian.

A continuación, se describen los sub-protocolos utilizados para cada una de las tareas.

Envío de credenciales

Una de las primeras fases de la comunicación es el envío de credenciales, el cual permite al servidor rechazar o aceptar las peticiones de conexiones entrantes. Para llevar a cabo esta discriminación de conexiones y realizar la distinción de dispositivos, se envía el NIA del alumno, el cual es comparado con la lista de alumnos importada en el servidor.

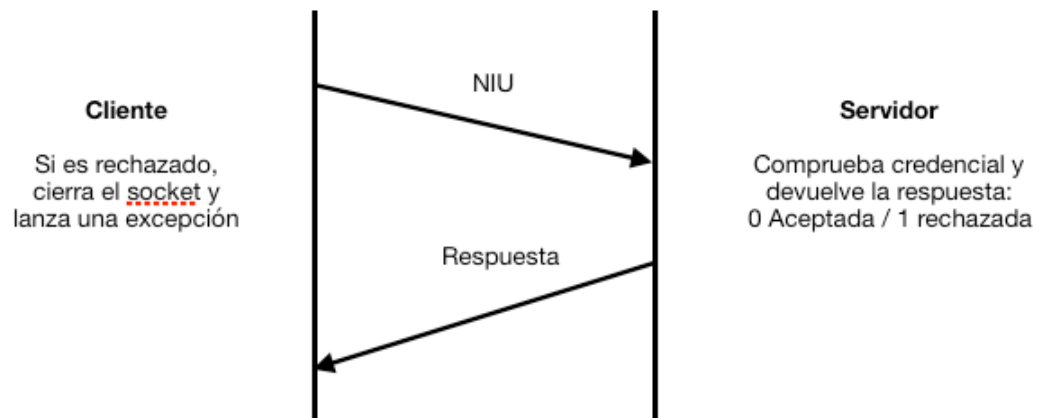


Ilustración 16 - Sub-protocolo envío credenciales

Envío de opciones

El proceso de comunicación entre cliente y servidor para el envío de las opciones:

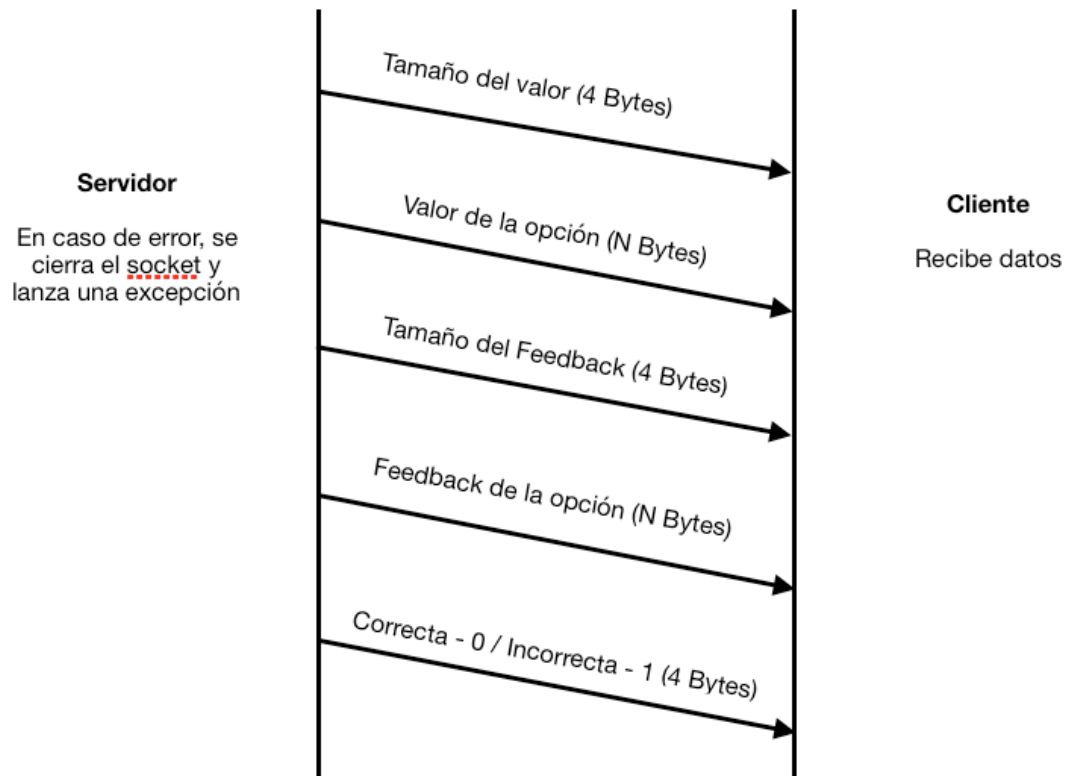


Ilustración 17 - Sub-protocolo envío de opciones

Envío de preguntas

El proceso de comunicación entre cliente y servidor para el envío de preguntas es el siguiente:

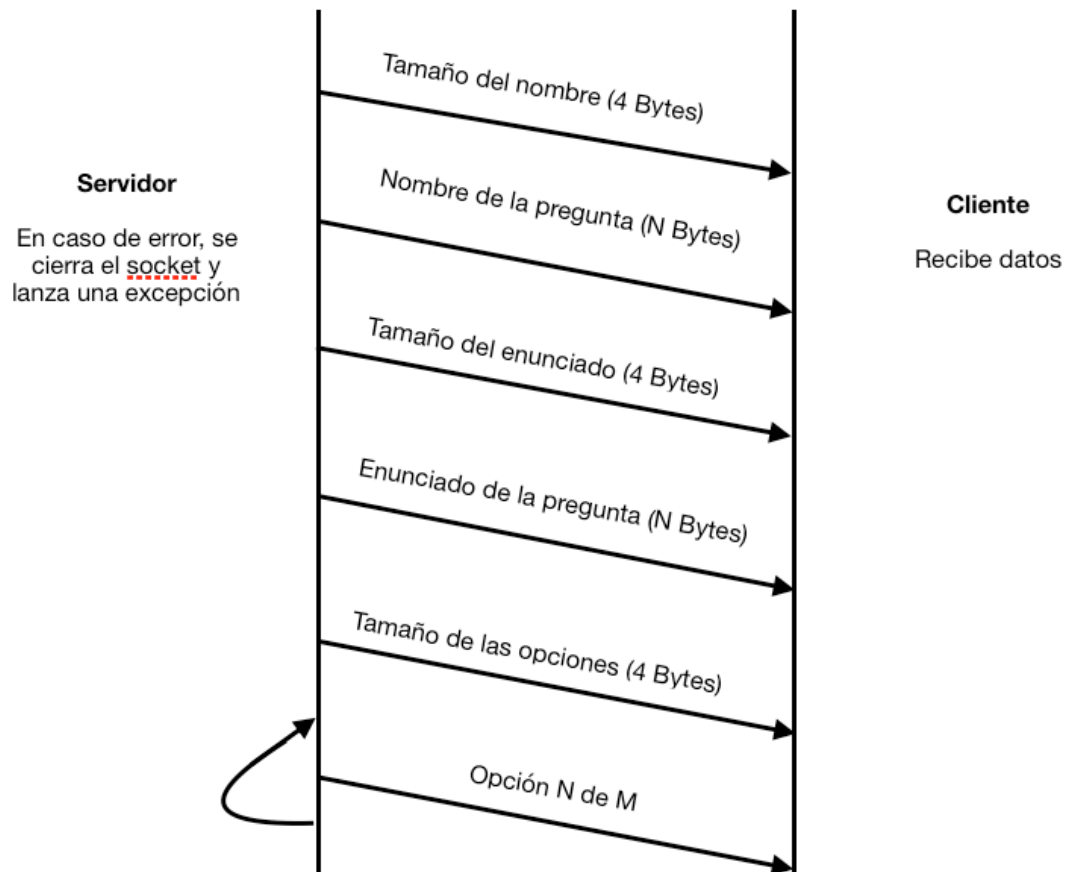


Ilustración 18 - Sub-protocolo envío de preguntas

Envío configuración de test

Dado que cada test es configurado por el profesor, la manera más eficaz de hacerlo es enviando una tabla hash, a continuación, se muestran los valores:

Posición	Objeto Representado	Valores	Significado
0	<i>optionType</i>	1 – Multi-respuesta	Permite seleccionar el tipo de preguntas
		0 – Mono-respuesta	
1	<i>answerType</i>	0 – Estándar	Lista de posibles opciones
2	<i>visualitation</i>	2 – Ambos	Establece el modo de visualización del enunciado de las preguntas
		1 – Solo Cliente	
		0 – Solo Servidor	
3	<i>feedbackType</i>	2 – Público	Establece el modo de visualización del feedback y las calificaciones
		1 – Anónimo	
		0 – Sin Feedback	
4	<i>orderType</i>	1 – Aleatorio	Establece el orden de las preguntas
		0 – En Orden	
5	<i>modifyAnswerEnabled</i>	1 – No	Determina si puede modificarse una respuesta
		0 – Si	
6	<i>time</i>	$1 \text{ a } 2^{31} - 1$	Tiempo máximo de respuesta para cada pregunta. En caso de estar fuera de rango su valor es 0
7			
8			
9			
10	<i>pass</i>	Cadena de 6 caracteres	Permite establecer una contraseña para acceder al test, en caso de que fuera necesario.
11			
12			
13			
14			
15			

Tabla 71 - Especificación Hash de configuración

Envío test completo

El proceso de comunicación entre cliente y servidor para el envío de preguntas es el siguiente:

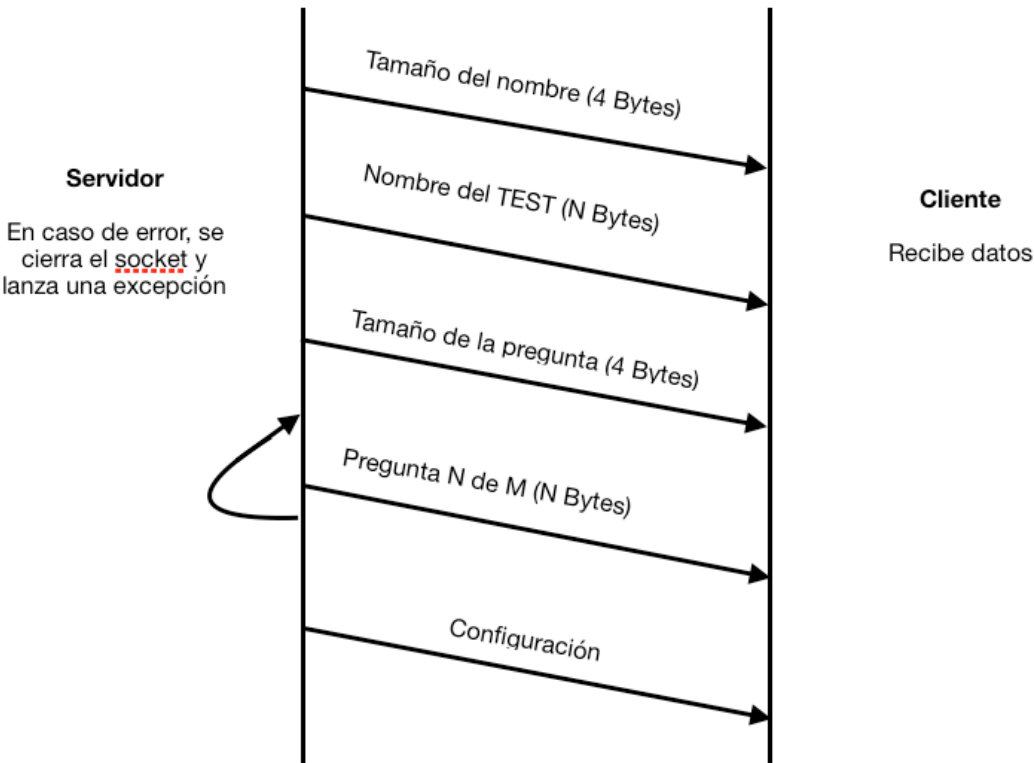


Tabla 72 - Sub-protocolo envío test completo

Funcionamiento del sistema

A continuación, se muestran los diagramas de secuencia que permiten describir el funcionamiento de la aplicación, mostrando las interacciones entre las clases, la interfaz y el usuario.

Caso de uso: Ejecutar TEST

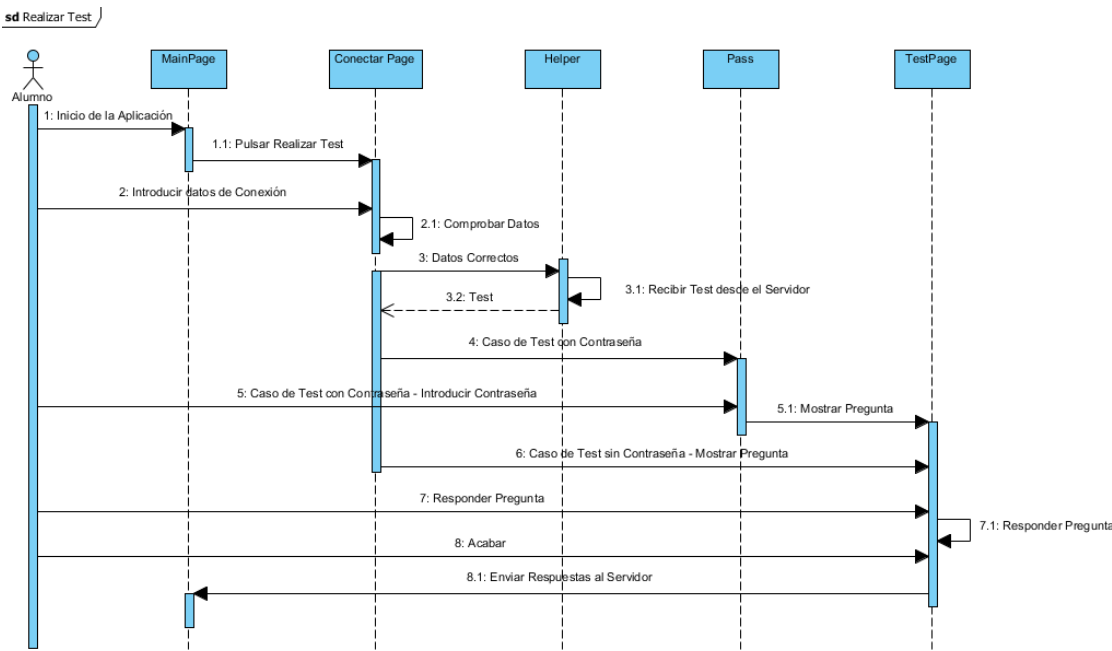


Ilustración 19 - Diagrama de secuencia - Ejecutar TEST

Caso de uso: Histórico de calificaciones

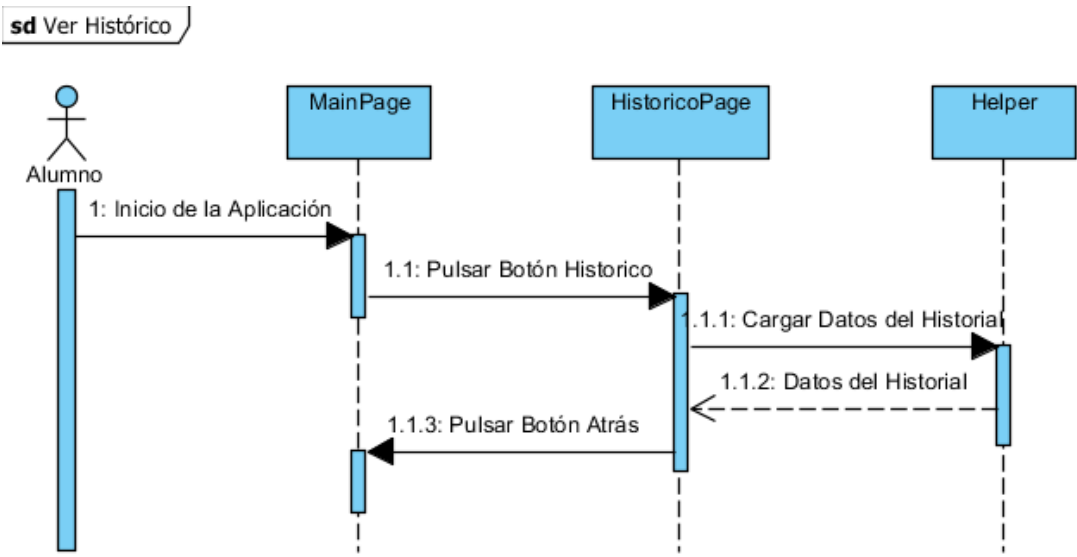


Ilustración 20 - Diagrama de secuencia - Histórico de calificaciones

Interfaces

A continuación, se muestran las interfaces de usuario a partir de las cuales se construirá la aplicación.

LaunchScreen



Ilustración 21 - Interfaz – LaunchScreen

Home

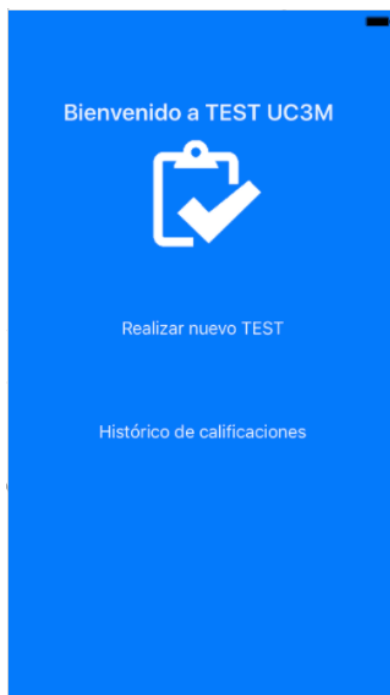


Ilustración 22 - Interfaz – Home

Conexión

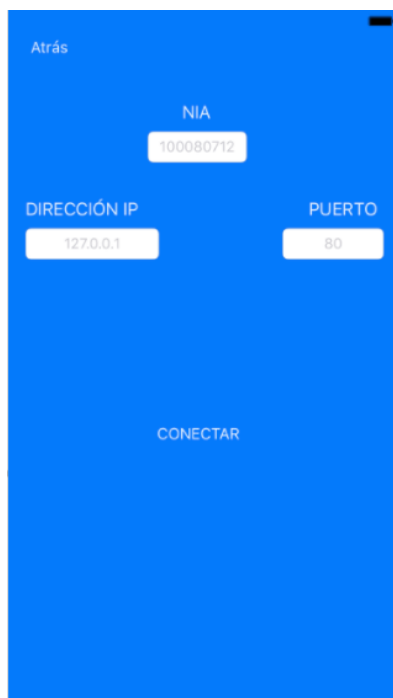


Ilustración 23 - Interfaz – Conexión

Test

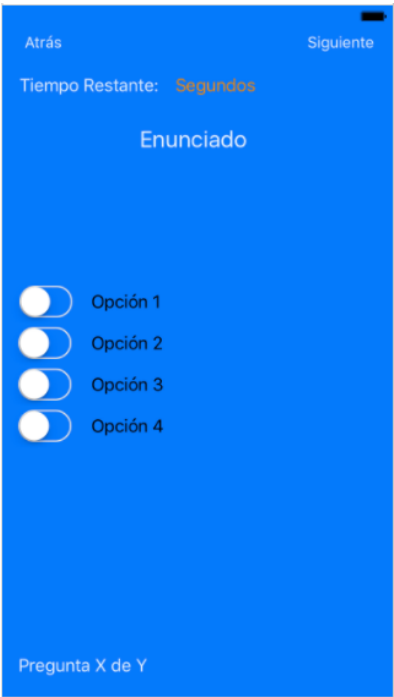


Ilustración 24 - Interfaz – Test

Histórico calificaciones

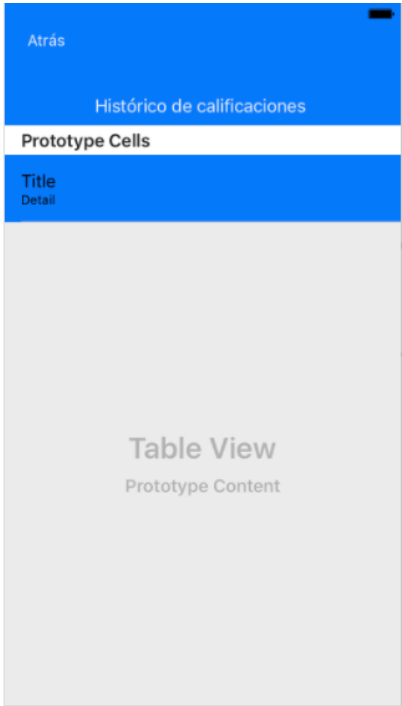


Ilustración 25 - Interfaz - Histórico calificaciones

Diagrama de navegabilidad



Ilustración 26 - Diagrama de navegabilidad

Implementación

Tras tener definidas tanto las interfaces como las funcionalidades, es el momento de comenzar con el desarrollo del proyecto. En este apartado se describen las fases del desarrollo, los problemas que han debido ser solventados y el plan de pruebas ejecutado.

Implementación del sistema

Para la implementación del sistema ha sido utilizada la plataforma de desarrollo Xcode en su versión 8.2, mediante el uso de ficheros “.swift” (para la lógica del sistema), ficheros “.storyboard” (para el diseño de la interfaz) y el SDK de iOS 10.

El equipo informático utilizado para el desarrollo de la aplicación ha sido el siguiente:

- MacBook Air (mediados de 2012)
- Procesador Intel Core i5 1.8 GHz
- Sistema Operativo macOS Sierra (versión 10.12.6)
- Memoria RAM de 4 GB DDR3 1600 MHz

Problemas encontrados

1. Lenguaje de programación inmaduro, durante el desarrollo de la aplicación y estudio de la API, la versión de la misma ha sido actualizada, cambiando un gran número de métodos y funciones propias del SDK, lo que ha conllevado continuas adaptaciones del código.
2. Concurrencia, hasta la versión 8 de Swift no está prevista introducir la concurrencia como parte de la API, por lo que, en la versión actual, utilizar concurrencia resulta tosco y complicado.
3. Entendimiento del protocolo de comunicación, dado que es un protocolo especialmente creado para esta solución, y que no hay mucha información al respecto, ha sido difícil comprender el funcionamiento y qué paquete de bytes se espera enviar y recibir, ha sido un arduo trabajo el conseguir comprenderlo del todo.

Implantación

Dado que el proyecto es en un sistema operativo propietario de Apple, la implantación se realiza a través del mercado de aplicaciones de Apple.

Lo primero que se necesita es una licencia como desarrollador de Apple para iOS. Una vez conseguida, se debe crear una entrada en la aplicación iTunes Connect, desde el cual importar el archivo “.ipa” (aplicación compilada desde Xcode). Una vez rellenos los formularios pertinentes, la aplicación quedará a la espera de que Apple la revise y se cerciore que se cumplen todas las normas y estándares que piden, momento en que estará lista para ser descargada desde el mercado de aplicaciones de Apple.

Evaluación

Para garantizar el correcto funcionamiento de todo producto de software, es necesario probar la aplicación, para así poder minimizar los errores que puedan no ser detectados antes de poner la aplicación en producción.

Se clasificarán dichas pruebas, según el nivel de detalle de cada una, en los siguientes tipos:

- Pruebas unitarias: Pruebas realizadas durante el desarrollo del producto, sirven para verificar el correcto funcionamiento a nivel método/clase, validando así el funcionamiento de cada uno de ellos de manera individual. Al no ser requeridas por el cliente, no se analizarán.
- Pruebas de integración: Tras la finalización de las pruebas unitarias, y comprobar que individualmente cada componente funciona correctamente, se debe verificar que el conjunto funciona correctamente. Al no ser requeridas por el cliente, no se analizarán.
- Pruebas de validación: Tras comprobar las pruebas unitarias y de integración, se deben verificar que los requisitos solicitados por el cliente y los que han añadido los desarrolladores se cumplen, y así verificar que el producto es realmente lo que el cliente necesitaba.
- Pruebas de sistema: Si el producto desarrollado debe integrarse con sistemas externos, estas pruebas son las encargadas de verificar el correcto funcionamiento de la integración. El objetivo de dichas pruebas son verificar la usabilidad, la robustez, la seguridad y el rendimiento.
 - Pruebas de prestaciones: Estas pruebas se centran en los requisitos catalogados como de recursos, midiendo el tiempo de respuesta, tamaño de aplicación, etc.
- Pruebas de regresión: En estas pruebas se tratan de descubrir agujeros de seguridad o funcionalidad (Bugs), comportamientos indeseados, etc.

Descripción de las pruebas

A continuación, se analizarán mediante tablas, las pruebas de validación, sistema y prestaciones.

Identificador		
Nombre:		Requisito:
Descripción:		
Procedimiento:		
Resultado:		

Tabla 73 – Plantilla pruebas

Como con cada plantilla de tabla, se procederá a la explicación de cada uno de los campos:

- **Identificador:** Nombre unívoco para cada una de las pruebas, la nomenclatura será la siguiente PRXXX-NN, dónde PR hace referencia a que se trata de una prueba, XX, será el tipo de prueba (VAL, SYS, RND) y NN será un número secuencial.
- **Nombre:** Valor especificativo de la prueba.
- **Descripción:** Resumen de la prueba a realizar.
- **Procedimiento:** Pasos a seguir para realizar la prueba.
- **Resultado:** Determina cual ha de ser el resultado a obtener, cualquier otro valor será un fallo.
- **Requisito** fuente: Indica el origen de la prueba, con el requisito relacionado.

Pruebas de validación

PRVAL-01		
Nombre:	Conexión	Requisito: RSFNC-012
Descripción:	Comprobación de que se realiza la conexión	
Procedimiento:	<ol style="list-style-type: none"> 1. Abrir la aplicación 2. Pulsar sobre el texto, realizar nuevo test 3. Introducir las credenciales 4. Pulsar conectar 	
Resultado:	Se mostrará un mensaje de conexión exitosa y esperará para recibir test	

Tabla 74 - PRVAL-01

PRVAL-02		
Nombre:	Histórico	Requisito: RSFNC-013
Descripción:	Comprobación del funcionamiento del histórico	
Procedimiento:	<ol style="list-style-type: none"> 1. Abrir la aplicación 2. Pulsar sobre el texto, histórico de resultados 	
Resultado:	Se mostrará la vista de histórico, con los resultados guardados	

Tabla 75 - PRVAL-02

Pruebas de sistema

PRSYS-01		
Nombre:	TEST	Requisito: RSFNC-001 al RSFNC-010
Descripción:	Comprobación del funcionamiento al realizar TEST	
Procedimiento:	<ol style="list-style-type: none"> 1. Abrir la aplicación 2. Pulsar sobre el texto, realizar nuevo test 3. Introducir las credenciales 4. Pulsar conectar 5. Esperar a TEST 6. Realizar TEST 7. Pulsar sobre finalizar Test 	
Resultado:	<ul style="list-style-type: none"> - En la aplicación de escritorio aparecerá la nota del alumno - En la aplicación de móvil, se volverá a la pantalla principal - No se han mostrado errores en ninguna de las aplicaciones 	

Tabla 76 - PRSYS-01

PRSYS-02		
Nombre:	Desconexión	Requisito: RSFNC-011
Descripción:	Comprobación de que se realiza la desconexión	
Procedimiento:	<ol style="list-style-type: none"> 1. Abrir la aplicación 2. Pulsar sobre el texto, realizar nuevo test 3. Introducir las credenciales 4. Pulsar conectar 5. Cerrar en cualquier momento la conexión 	
Resultado:	Se mostrará un mensaje de desconexión que se verá si solo se ha enviado a background la aplicación	

Tabla 77 - PRSYS-02

Pruebas de rendimiento

PRRND-01		
Nombre:	Tamaño del ejecutable	Requisito: RSRSC-003
Descripción:	Comprobación de que el “.ipa” no ocupa más de 20MB	
Procedimiento:	Buscar la aplicación en el App Store y comprobar su tamaño	
Resultado:	N/A	

Tabla 78 - PRRND-01

Trazabilidad entre pruebas y requisitos

	PRVAL-01	PRVAL-02	PRSYS-01	PRSYS-02	PRRND-01
RSFNC-001			X		
RSFNC-002			X		
RSFNC-003			X		
RSFNC-004			X		
RSFNC-005			X		
RSFNC-006			X		
RSFNC-007			X		
RSFNC-008			X		
RSFNC-009			X		
RSFNC-010			X		
RSFNC-011				X	
RSFNC-012	X				
RSFNC-013		X			
RSFNC-014					
RSFNC-015					

Tabla 79 - Trazabilidad pruebas y requisitos

Planificación y presupuesto

Planificación

Como se explicó en el apartado de planificación, al inicio del documento, el compromiso adquirido con la entrega de una planificación detallada es de suma importancia.

Al ser una de las partes más importantes del proyecto, es esencial tener una buena documentación de cada fase, grabar correctamente las fechas de inicio y fin reales, a fin de comprobar el cumplimiento de la estimación, tanto en retrasos como adelantos o entregas a tiempo.

En este proyecto no se han sufrido retrasos considerados, la única discrepancia ha sido la finalización de la memoria, cuya fecha de entrega se adelantó 3 días puesto que se finalizó antes.

Presupuesto

A continuación, se detallan los costes asociados al desarrollo de la aplicación y los cálculos necesarios para obtenerlos.

Costes de recursos humanos

Para este proyecto se ha tenido en cuenta un equipo estándar de desarrollo de software, compuesto por un Jefe de Proyecto, un Analista, un técnico de pruebas y un desarrollador.

Según lo publicado en el BOE a fecha de 18 de enero de 2017 la tabla salarial para cada empleado y los costes asociados, sin tener en cuenta experiencia ni grado de conocimiento (junior/senior) serán los siguientes:

Rol	Coste (€) por hora	Total dedicación (h)	Coste (€) imputable
Jefe de proyecto	13,12	50	656
Analista	9,74	100	974
Técnico de pruebas	9,39	50	469,5
Desarrollador	7,50	100	750
TOTAL			2849,5

Tabla 80 - Costes recursos humanos

Coste de licencia

Otro gasto a tener en cuenta es el de las licencias de software, teniendo en cuenta que existen diferentes tipos de licenciamiento, quedarán recogidos en el siguiente formulario.

Como no tiene sentido cobrar el total de una licencia anual o de un programa que va a ser usado, se imputará únicamente el precio de amortización, teniendo en cuenta que la vida útil suelen ser 5 años. En la siguiente tabla quedará reflejado:

Descripción	Coste (€)	Dedicación (meses)	Periodo de depreciación	Coste imputable
macOS Sierra	0,00	2	60	0,00
Office	149,00	2	12	25
Licencia desarrollo iOS	99,00	2	12	16,50
Gantt Project	0,00	2	60	0,00
Xcode	0,00	2	60	0,00
iOS	0,00	2	60	0,00
TOTAL				41,5

Tabla 81 - Costes de software

El coste imputable se ha calculado dividiendo el coste total entre el periodo de depreciación y multiplicando este resultado por el número de meses utilizado.

Costes de hardware

Para poder utilizar el software anteriormente mencionado, es necesario disponer de dispositivos físicos. A continuación, se muestra una tabla con el desglose de estos gastos.

Descripción	Coste (€)	Dedicación (meses)	Periodo de depreciación	Coste imputable
MacBook Air	1200	2	60	40,00
IPhone 6 Plus	1000	2	60	33,5
TOTAL				73,5

Tabla 82 - Costes hardware**Costes indirectos**

Por último, aparte de los costes tangibles y de software, han existido los siguientes gastos indirectos:

Descripción	Coste (€) absoluto/mes	Dedicación (meses)	Coste imputable
Luz	30	2	60,00
Desplazamientos	180	2	360,00
Internet	40	2	80,00
TOTAL			500,00

Tabla 83 - Costes indirectos

Costes totales

Por lo tanto, los costes del proyecto son los siguientes:

Precio total del proyecto	
Coste de recursos humanos	2849,5
Coste de hardware	73,5
Coste de software	41,5
Costes indirectos	500
SUB-TOTAL	3464,5
Margen de beneficios	15%
TOTAL	4000

Tabla 84 - Costes totales

Así y con todo, el presupuesto a presentar al cliente, del proyecto que describe el documento asciende a **CUATRO MIL EUROS** (4000€) IVA no incluido.

Conclusiones y trabajos futuros

Finalmente se expondrán una serie de conclusiones, valoración personal del autor del proyecto y una serie de trabajos futuros con la finalidad de poder hacer evolutivos sobre el proyecto que mejoren la eficiencia, rendimiento y capacidades de la aplicación.

Conclusiones y valoración personal

El objetivo de la realización de este proyecto era crear una aplicación para dispositivos iOS para la realización de exámenes presenciales en la Universidad Carlos III de Madrid.

De este modo, se pretendía mejorar el método de evaluación y la participación de los alumnos en clase, haciéndoles partícipes en el proceso de evaluación. Haciendo uso de los Smartphones, dada su creciente evolución, y dejando de lado antiguos métodos, como los exámenes en papel físico, los cuales necesitan mucho tiempo para corregir u otros métodos, como los mandos inalámbricos, que además de ser caros, suponen llevar un montón de artilugios a clase.

El sistema se ha desarrollado de manera completa, cumpliendo de una forma u otra con los requisitos, teniendo en cuenta que Swift aun no es un lenguaje maduro, y que muchas de las funcionalidades internas están aún por implementar, a pesar de esto, es el lenguaje por el que apuesta Apple, y por el que está incentivando a los desarrolladores, prestando cada vez menos apoyo a Objective C-y ampliando cada vez más las comunidades de desarrolladores Swift

Habiendo cumplido todos los requisitos solicitados, la aplicación queda lista, para que cuando se considere necesario, se realice la petición de subida a la App Store.

Personalmente este proyecto me resultaba interesante puesto que siempre quise realizar alguna aplicación móvil. Tuve que decidir que lenguaje de programación usar, entre Objective C y Swift, me decidí por este último por ser un lenguaje novedoso, el cual me tenía intrigado.

Sin embargo, considero que el proceso de aprendizaje del nuevo lenguaje no fue complicado, en parte gracias a los profundos conocimientos adquiridos gracias a las

asignaturas cursadas bajo el Plan de informática 2008. A continuación, detallo las asignaturas que, a mi parecer, han tenido gran importancia para la consecución de este proyecto:

- Programación: Es la asignatura por excelencia de la carrera, la que nos enseñó a programar, es el comienzo del pensamiento y razonamiento informático.
- Álgebra lineal: El uso de matrices es esencial para el tratamiento de datos.
- Estructura de datos y algoritmos: Gracias a esta asignatura es más simple comprender como estructurar la información contenida en el sistema a desarrollar y seleccionar la mejor manera de realizarlo.
- Técnicas de búsqueda y uso de la información: Esta asignatura te enseña a desenvolverte en internet, a buscar correctamente sin perder tiempo.
- Principios de la ingeniería informática: Esta asignatura ayuda a comprender como afrontar cada parte del desarrollo, a desmenuzar un ente complejo para así alcanzar la meta final.
- Técnicas de verificación del software: Conocer los límites del sistema y tener controlado cada punto del mismo para evitar todos los posibles errores que puedan suponerse de ellos, ayuda enormemente a saber analizar las pruebas sobre el software, a documentarlos y ejecutarlos
- Sistemas operativos: En esta asignatura se introdujo el concepto de hilo, de procesos ligeros y pesados y cómo manejarlos para que la eficiencia del sistema sea mayor
- Redes de ordenadores: En esta asignatura se nos enseñó el comportamiento del mundo de la red, direccionamiento ip, capas de servicio, comunicaciones, etc.
- Interfaces de usuario: Toda la teoría dada en esta asignatura, aunque en gran parte estaba enfoca a desarrollo web, no difiere demasiado en el desarrollo móvil, tener claros los principios de Nielsen, colores, formas, lugares, buenas prácticas, etc.
- Procesos de desarrollo del software: Esta asignatura sí detalla en gran medida todo el ámbito de la Ingeniería del Software acerca de la toma requisitos y la asignación y división de funcionalidades en un proyecto.

- Arquitectura de computadores: Los paralelismos, y los conceptos de hilos introducidos en sistemas operativos, se acaban de asentar con esta asignatura.
- Dirección de proyectos de desarrollo de software: Esta asignatura es el sumun de la ingeniería de software, puesto que trata de simular una empresa que lleva a cabo un proyecto de software con, absolutamente toda la documentación relativa al mismo.

Conseguir finalizar este proyecto, ha hecho que me vea capaz de realizar cualquier desarrollo de software, sentirme preparado para lo que pueda encontrarme en el mundo laboral y prepararme para afrontar cada uno de los problemas que pueda surgirme en el proceso.

Trabajos futuros

Como trabajo futuro queda la implementación total con Moodle, puesto que no ha podido ser llevada a cabo por un requisito negativo del cliente, ya que el departamento que debía darle permisos se lo negó, por ello, hubo que buscar una alternativa.

También quedaría poder añadir algo que ahora está muy de moda, procesamiento del lenguaje natural, con esto, se podría ampliar el rango de actuación de la aplicación fuera de los simples test.

También se podría implementar una aleatoriedad a la hora de pintar las respuestas, de este modo se dificulta que los alumnos puedan copiar entre ellos.

Anexo I

Manual usuario aplicación

A continuación, se muestra un manual de uso de la aplicación. Como se podrá comprobar el uso es fácil e intuitivo.

Los elementos gráficos pueden variar respecto a la versión final del producto.

Visualizar histórico

Para llegar a la vista del histórico de calificaciones se deben seguir los pasos descritos a continuación:

1. Iniciar la aplicación.
2. Pulsar sobre Histórico de calificaciones.

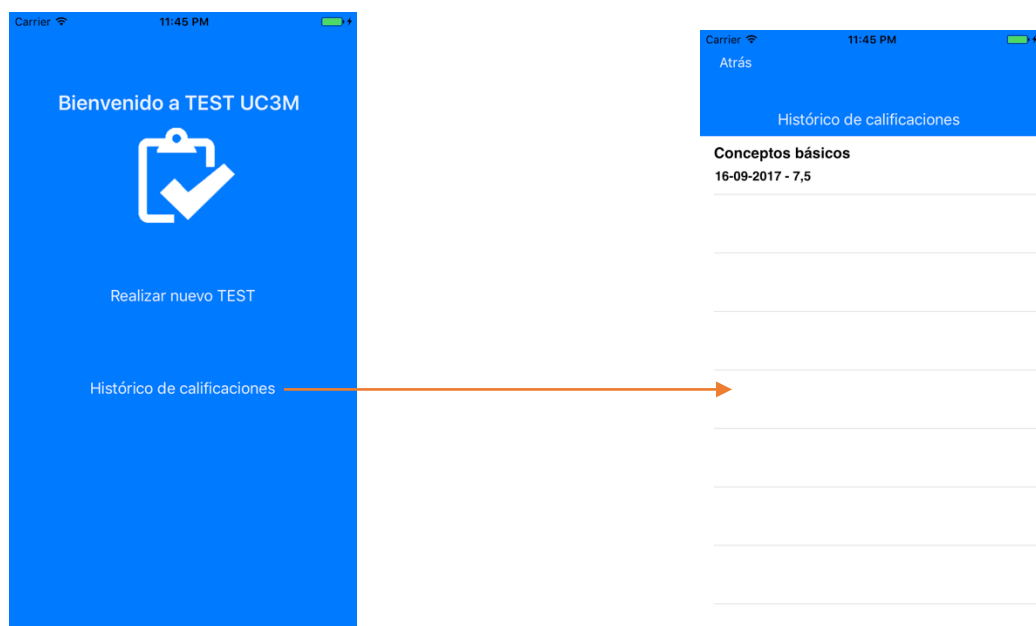


Ilustración 27 - Manual Histórico resultados

Realizar test

Los pasos a seguir serán los siguientes:

1. Iniciar la aplicación.
2. Pulsar sobre Realizar nuevo TEST.
3. Introducir los datos de conexión.

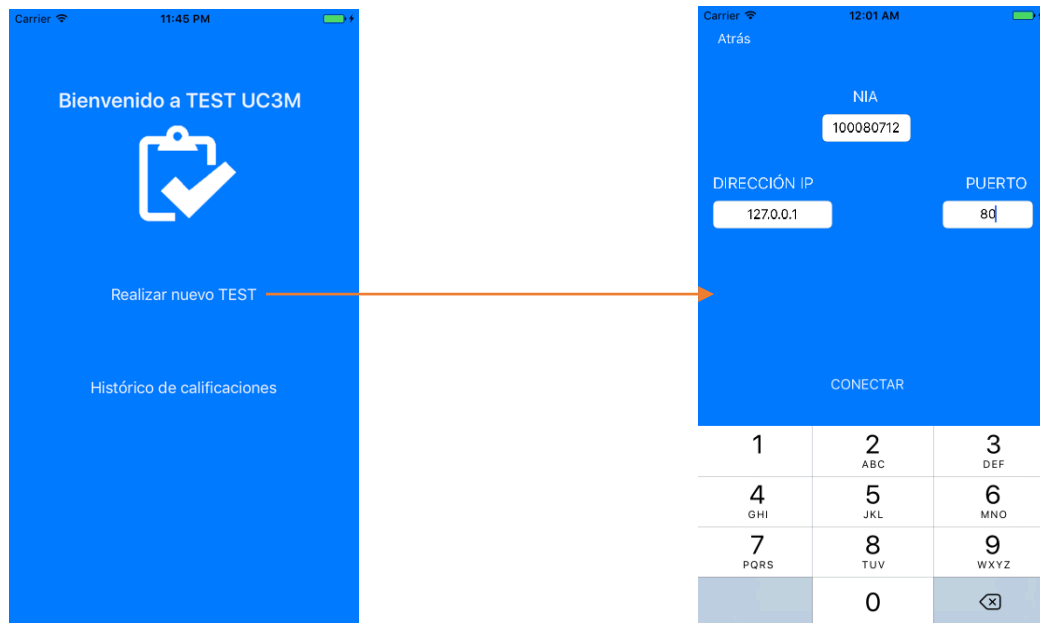


Ilustración 28 - Manual Conexión

4. Si algún dato introducido fuese incorrecto, aparecerá un mensaje de error.
 - a. En caso de que el test estuviese protegido con contraseña, se introducirá:

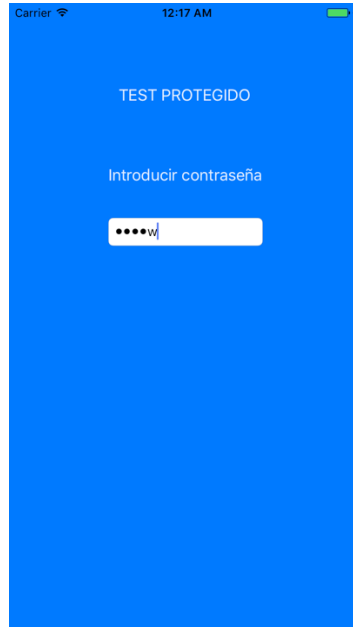


Ilustración 29 - Manual Contraseña

5. Una vez el profesor de permiso se mostrará la pantalla de test y dará inicio:

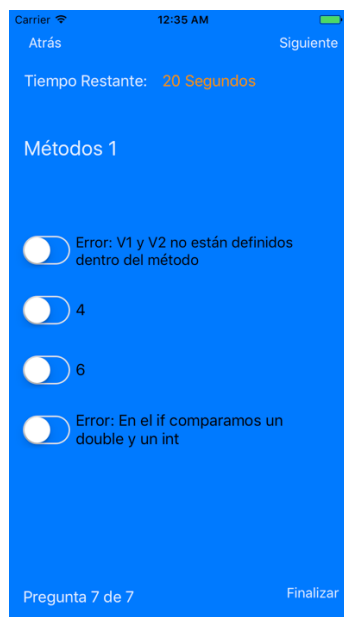


Ilustración 30 - Manual TEST

6. Una vez finalizado el test, se pulsa sobre finalizar:

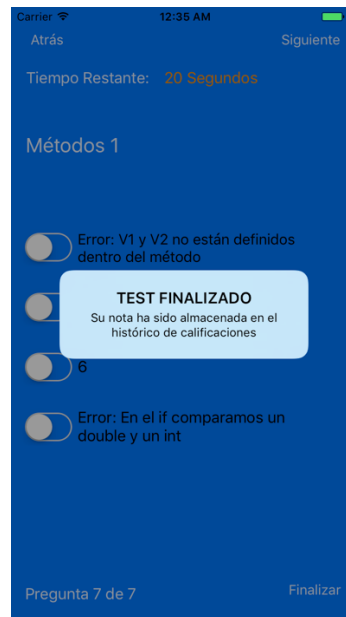


Ilustración 31 - Manual finalización

Referencias

Bibliografía

- [1] iOS developer program [En línea] Recuperado el 10 de julio de 2017, <https://developer.apple.com/account/#/overview/QUX8DU2B6G>
- [2] The Swift Programming Language (Swift 4) [En línea] Recuperado el 10 de julio de 2017, en iBooks
- [3] StackOverflow [En línea] múltiples consultas, <https://stackoverflow.com/questions/tagged/swift>
- [4] Características y tabla comparativa de los sistemas móviles más usados. [En línea] Recuperado el 20 de agosto de 2017, <http://jmacuna73.blogspot.com.es/2017/03/sistemas-operativos-moviles.html>
- [5] Android 8, [En línea] Recuperado el 22 de agosto de 2017, <https://www.android.com/versions/oreo-8-0/>
- [6] iOS 11, [En línea] recuperado el 12 de septiembre de 2017, <https://www.apple.com/es/ios/ios-11/>
- [7] Cuota de mercado móvil en España, [En línea] recuperado el 21 de agosto de 2017, <http://es.kantar.com/tech/móvil/2017/agosto-2017-cuota-de-mercado-de-smartphones-en-españa-mayo-2017/>
- [8] Guía para la realización de un proyecto de investigación científica, [En línea] recuperado el 17 de julio de 2017, http://www3.uc3m.es/reina/ENVIOS/Idioma_1/2014/231.15183.pdf
- [9] Boletín oficial del estado, tablas salariales, [En línea] recuperado el 15 de septiembre de 2017, <https://www.boe.es/boe/dias/2017/01/18/pdfs/BOE-A-2017-542.pdf>
- [10] Documentación para Métrica v.3, [En línea] Recuperado el 2 de agosto de 2017, http://administracionelectronica.gob.es/pae_Home/pae_Documentacion/pae_Metodolog/pae_Metrica_v3.html#.VODPgS5-bzI
- [11] GanttProject, [En línea] Recuperado el 20 de julio de 2017 <http://www.ganttproject.biz>

- [12] Programa Grado en Ingeniería Informática (2008). (24 de 03 de 2009).
Recuperado el 18 de julio de 2017, de Universidad Carlos III de Madrid:
http://portal.uc3m.es/portal/page/portal/titulaciones_grado/infor/plan_2008

Glosario de términos

- **App**, (del inglés application), tipo de programa informático diseñado como herramienta para permitir a un usuario realizar diversos trabajos.
- **Bug**, error/fallo en un programa informático que desencadena un acontecimiento indeseado
- **Endianess**, designa el formato en el que se almacenan los datos de más de un byte en un ordenador.
- **Feedback**, retroalimentación, es un mecanismo por el cual una cierta información de salida de un sistema se redirige a la entrada, con el objeto de controlar us comportamiento.
- **HTTP**, (HyperText Transfer Protocol) protocolo utilizado en cada transacción de la web.
- **IDC**, (International Data Corporation) es una asesoría situada en Estados Unidos, especializada en el análisis del mercado en ámbito de las tecnologías de la información, las telecomunicaciones y la tecnología del consumo.
- **IP**, (internet protocol) protocolo de comunicación de datos clasificado funcionalmente en la capa de red según el modelo de datos internacional.
- **LMS**, (Learning Management System) es un software instalado en un servidor que se emplea para administrar, distribuir y controlar las actividades de formación no presencial de una institución u organización
- **Marshalling**, serialización, consiste en un proceso de codificación de un objeto en un medio de almacenamiento, con el fin de transmitirlo a través de una conexión en red como una serie de bytes o en un formato legible por un humano.
- **PC**, ordenador personal, es un tipo de computador diseñada para ser utilizado por una persona.

- **SDK**, (del inglés Software Development Kit) es un conjunto de herramientas de desarrollo de software que le permite al programador crear aplicaciones para un sistema concreto.
- **Smartphone**, es un tipo de teléfono móvil construido sobre una plataforma móvil informática, semejante a un ordenador.